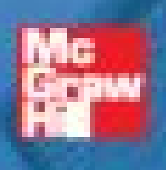
SEGUNDA EDICIÓN

FUNDAMENTOS DE ROBÓTICA

Antonio Barrientos
Luis Felipe Peñín
Carlos Balaguer
Rafael Aracil



FUNDAMENTOS DE ROBÓTICA

Antonio Barrientos
Luís Felipe Peñín
Carlos Balaguer
Rafael Aracil

Universidad Politécnica de Madrid

McGraw-Hill

MADRID • BUENOS AIRES • CARACAS • GUATEMALA • LISBOA • MÉXICO • NUEVA YORK
PANAMÁ • SAN JUAN • SANTAFÉ DE BOGOTÁ • SANTIAGO • SÃO PAULO
AUCKLAND • HAMBURGO • LONDRES • MILÁN • MONTREAL • NUEVA DELHI • PARÍS
SAN FRANCISCO • SIDNEY • SINGAPUR • ST. LOUIS • TOKIO • TORONTO

FUNDAMENTOS DE ROBÓTICA

McGRAW-HILL/INTERAMERICANA DE ESPAÑA, S. A.
Edificio Valrealty
Basauri, 17, 1.ª planta
28023 Aravaca (Madrid)

ISBN: 84-481-0815-9
Depósito legal: M. 2.825-1997

Editora: Concepción Fernández Madrid
Cubierta: Estudio F. Piñuela
Foto de cubierta: Cortesía de DISAM/UPM
Compuesto por los autores
Impreso en: Lavel, S. A.

IMPRESO EN ESPAÑA - PRINTED IN SPAIN

Acerca de los autores

Antonio Barrientos Cruz es Ingeniero Industrial especialidad Automática y Electrónica por la Escuela Técnica Superior de Ingenieros Industriales de Madrid (1982) y Doctor Ingeniero Industrial por la Universidad Politécnica de Madrid (1986). Desde 1988 es profesor titular del Departamento de Automática Ingeniería Electrónica e Informática Industrial de la Universidad Politécnica de Madrid, impartiendo entre otros cursos los correspondientes a regulación automática, robótica, computadores y automatización de la producción. Desde 1984 ha impartido en múltiples ocasiones cursos de robótica, tanto como parte de los estudios de Ingeniero Industrial como dentro de diferentes programas para el sector industrial.

Luis Felipe Peñín Honrubia es Ingeniero Industrial especialidad Automática Electrónica por la Universidad Politécnica de Madrid. En la actualidad es investigador del grupo de robótica del Departamento de Automática, Ingeniería Electrónica e Informática Industrial (DISAM) de la Universidad Politécnica de Madrid, habiendo impartido diversos cursos de automatización y robótica tanto como parte de los estudios de Ingeniero Industrial como en diferentes programas para el sector industrial.

Carlos Balaguer Bernaldo de Quirós obtuvo el grado de Ingeniero Electromecánico por el Instituto Energético de Moscú (1977) e Ingeniero Industrial especialidad Automática por la Universidad Politécnica de Madrid (1981). Doctor Ingeniero por la Universidad Politécnica de Madrid (1983), ha sido profesor titular del Departamento de Automática, Ingeniería Electrónica e Informática Industrial impartiendo cursos de control de procesos y robótica. En la actualidad es Catedrático del Departamento de Ingeniería Eléctrica en la Universidad Carlos III de Madrid.

Rafael Aracil Santonja es Ingeniero Industrial especialidad Eléctrica y Doctor Ingeniero Industrial por la Universidad Politécnica de Madrid. En 1973 se incorporó al Departamento de Automática, Ingeniería Electrónica e Informática Industrial de la Universidad Politécnica de Madrid donde, desde 1983, es Catedrático. En la actualidad coordina las áreas de robótica y visión artificial.

Contenido

Prefacio

En menos de 30 años la robótica ha pasado de ser un mito, propio de la imaginación de algunos autores literarios, a una realidad imprescindible en el actual mercado productivo.Tras los primeros albores, tímidos y de incierto futuro, la robótica experimentó entre las décadas de los setenta y ochenta un notable auge, llegando en los noventa a lo que por muchos ha sido considerado su mayoría de edad, caracterizada por una estabilización de la demanda y una aceptación y reconocimiento pleno en la industria.

La formación del profesional de la ingeniería, tanto en sus ramas de automatización, mecánica o incluso generalista, no ha podido dejar de lado esta realidad y ha incluido desde finales de los ochenta a la robótica como parte de sus enseñanzas.

La robótica posee un reconocido carácter interdisciplinario, participando en ella diferentes disciplinas básicas y tecnologías tales como la teoría de control, la mecánica, la electrónica, el álgebra y la informática, entre otras. Numerosas obras, la mayor parte en inglés, han surgido en el mercado bibliográfico desde 1980. En algunas de ellas se aprecia la tendencia de presentar la robótica como un simple repaso de tecnologías básicas (electrónica, informática, mecánica, etc.). En otros, por el contrario, se abusa de una excesiva especialización, bien limitando el estudio de la robótica a un complicado conjunto de problemas físico matemáticos, orientados a conocer con detalle el funcionamiento del sistema de control o bien tratando la robótica únicamente desde el punto de vista del usuario, sin dar una adecuada información sobre las tecnologías que la conforman.

Ninguno de los planteamientos anteriores es a nuestro juicio acertado. Una formación en robótica focalizada exclusivamente en el control de robots no es la más útil para la mayoría de los estudiantes, que de trabajar con robots lo harán como usuarios y no como fabricantes. Sin embargo, no hay que perder de vista que se está formando a ingenieros, y que hay que proveerles de los medios adecuados para abordar, de la manera más adecuada, los problemas que puedan surgir en el desarrollo de su profesión.

Por estos motivos, en este texto se ha procurado llegar a un adecuado equilibrio entre los temas relacionados con el conocimiento profundo del funcionamiento de un robot (en sus aspectos mecánico, informático y de control) y aquellos en los que se proporcionan los criterios para evaluar la conveniencia de utilizar un robot y el modo más adecuado de hacerlo.

El libro está fundamentalmente dirigido al estudiante de ingeniería en sus especialidades de Automatización, Electrónica o similar. Con su estudio el alumno adquirirá los conocimientos necesarios para poder abordar adecuadamente el proyecto de una instalación robotizada, junto con una detallada comprensión del funcionamiento del robot. Asimismo, su lectura es aconsejable para los estudiantes de ingeniería técnica o superior de cualquier otra especialidad.

El enfoque eminentemente práctico y aplicado de buena parte del texto, lo hace también útil para todo aquel profesional que desee adquirir los conocimientos necesarios para entender y utilizar adecuadamente la robótica.

Su contenido es el de un primer curso de robótica, que se considera completo para todo aquel que quiera iniciarse en la misma sin dejar de lado sus múltiples vertientes. Es a su vez imprescindible para aquellos que pretendan seguir sus estudios de posgrado dentro del área de la robótica, en sus aspectos más complejos e innovadores.

Para su correcta comprensión es preciso una formación básica en ingeniería. Siendo en concreto deseable tener una adecuada base en álgebra matricial, electrónica, control e informática (programación, arquitectura de computadores). No obstante, cualquier otro lector sin esta formación específica, podrá leer sin dificultad varios de los capítulos del libro, y captar, en los capítulos más complejos, cuáles son los objetivos sin tener que ahondar en los medios usados para lograrlos.

El libro se ha organizado de acuerdo a la experiencia docente de los autores, que suman más de 30 años impartiendo cursos de robótica dentro del marco académico, así como en cursos de especialización para la industria. Se ha procurado el adecuado equilibrio entre los capítulos dedicados al control de robot (Capítulos 3,4,5,6,7) con los dedicados a su implantación y uso (Capítulos 2,8,9,10), incluyéndose en todos un adecuado número de ejemplos que faciliten el estudio y comprensión.

En el Capítulo 1 se presenta la robótica como tecnología interdisciplinar, definiendo al robot industrial y comentando su desarrollo histórico y estado actual.

El Capítulo 2 está dedicado al estudio de los elementos que componen un robot: estructura mecánica, transmisiones y reductores, actuadores, elementos terminales y sensores. No se pretende en el mismo hacer un estudio exhaustivo de estos componentes, pues no es este objetivo propio de la robótica, sino destacar las características que los hacen adecuados para su empleo en los robots.

Para que un robot pueda moverse y manipular objetos, es necesario una adecuada localización de las configuraciones espaciales por las que se pretende que el robot pase. Tanto para el usuario del robot como para su diseñador es necesario manejar adecuadamente una serie de herramientas matemáticas que permitan situar en posición y orientación un objeto (en particular, el extremo del robot) en el espacio tridimensional. El Capítulo 3 recoge de manera pormenorizada las diferentes alternativas existentes para tal fin. Su adecuada asimilación es imprescindible para abordar tanto los capítulos de control como el de programación.

El estudio de la cinemática del robot permite relacionar la posición de sus actuadores con la posición y orientación del extremo. Esta relación no trivial, estudiada en el Capítulo 4, es fundamental para desarrollar el control del robot.

El Capítulo 5 aborda el modelado dinámico de un robot. Como sistema dinámico, el robot es uno de los sistemas más fascinantes para el control, por incorporar muchas de aquellas dificultades que clásicamente se obvian en el estudio del control de sistemas. Por este motivo, el Capítulo 5, junto con el 7, además de su indudable interés para el estudio de la robótica, constituye un excelente colofón para los estudios de control avanzado propios de un ingeniero de automatización.

Tras establecer los modelos cinemático y dinámico de un robot en los Capítulos 4 y 5, respectivamente, se está en disposición de abordar el control del mismo. En el Capítulo 6 se trata el control cinemático, cuyo objetivo es definir con precisión las trayectorias temporales que deberán desarrollar cada uno de los actuadores del robot para conseguir los objetivos de velocidad y precisión de la mejor manera posible. El resultado de esta definición de trayectorias debe ser llevado a la práctica por el control dinámico, objeto del Capítulo 7, que trabajando con estrechas limitaciones de tiempo de cál-

culo sobre un sistema multivariable, fuertemente no lineal y de parámetros variables, debe conseguir elevadas prestaciones.

Tras el paso por el modelado y control del robot, se abordan los aspectos más prácticos de la robótica, comenzando en el Capítulo 8 por la programación. Esta es tratada en primer lugar desde un punto de vista general, analizando las características de los diferentes procedimientos de programación de robots existentes. En segundo lugar se presentan dos de los lenguajes de programación de robots más extendidos: el V+ (Adept) y el más reciente RAPID (ABB).

A la hora de llevar a la práctica la implantación de un sistema robotizado, el ingeniero debe tener criterio para seleccionar cuál es el robot más adecuado para su aplicación, así como poder evaluar su beneficio económico. Además, existen una serie de aspectos prácticos, como el diseño de la célula o la seguridad de la misma, sobre los que si bien es difícil establecer una metodología, sí es posible dar unas premisas básicas que sirvan de guía y orientación a quien se enfrenta por primera vez con este tipo de problemas. En el Capítulo 9 se tratan de cubrir estos aspectos prácticos, incorporando además un breve repaso al mercado actual de la robótica industrial.

El último capítulo está dedicado a la utilización del robot en diferentes tipos de aplicaciones, tanto las más habituales, como la soldadura de carrocerías, como las más novedosas en los sectores de la construcción o cirugía. Es evidente que la robótica está siendo aplicada a un gran número de sectores dispares, siendo imposible pretender que el estudiante de la robótica conozca las características de los mismos (lo mismo cabría decir para el estudiante de la automatización de procesos en general). No obstante, tras el estudio del Capítulo 10 el alumno tendrá una idea formada sobre dónde se están aplicando los robots, los motivos por los que su empleo es rentable y, lo que puede ser más importante, una adecuada capacidad para proyectar estas soluciones robotizadas estudiadas a cualquier otra aplicación o sector.

El contenido total del libro puede, a juicio de los autores, constituir un curso básico de robótica de una duración cuatrimestral, aproximadamente.

Los autores quisiéramos agradecer muy sinceramente, la ayuda y el apoyo recibido en todo momento de los profesores, ingenieros e investigadores, que han formado parte de DISAM y en particular de su grupo de robótica: Francisco José Rodríguez, Hortensia Amarís, José Manuel Pastor, Ernesto Gambao, Carlos Francisco Rodríguez, Roque Saltarén, Andrés García, Manuel Ferre, Antonio Giménez y Jesús Carrera, ahora profesores de otras Universidades. Todos ellos han compartido con nosotros las penas y alegrías que conlleva la investigación y la enseñanza de la robótica. En este libro hay, sin duda alguna, muchos "sacos de arena" aportados por ellos.

Todos los profesores y miembros de DISAM han contribuido, durante 15 largos años, a que existan los medios necesarios para poder desarrollar las múltiples investigaciones y desarrollos que han permitido a los autores adquirir los conocimientos que se reflejan en esta obra. A todos ellos nuestro reconocimiento.

También debemos agradecer el esfuerzo realizado en la revisión y corrección del texto por Antonio Giménez, Miguel Hernando y Javier Fernández-Pello, así como a Eduardo Avanzini, autor de las recreaciones de robots del Capítulo 1.

De manera muy especial agradecemos a Jesús Carrera su participación destacada en el desarrollo de algunos de los capítulos, y a nuestra editora, Concha Fernández, que nos ha apoyado y ha confiado en nosotros.

Y como no, queremos recordar a nuestras familias y seres queridos, a quienes agradecemos su comprensión por las horas robadas y dedicamos, con todo nuestro cariño, el resultado de un largo esfuerzo.

Por último, este libro está escrito pensando en nuestros alumnos, pasados, presentes y futuros. Confiamos en que les sea útil y que aporte algo en su formación. Con nuestro mayor respeto hacia ellos...

Madrid, noviembre de 1996
Los autores

Capítulo 1

INTRODUCCIÓN

Antes que acabes, no te alabes
Refranero popular

El ciudadano industrializado que vive a caballo entre el siglo XX y el XXI se ha visto en la necesidad de aprender, en escasos 25 años, el significado de un buen número de nuevos términos marcados por su alto contenido tecnológico. De ellos sin duda el más relevante haya sido el del ordenador (o computador). Éste, está introducido hoy en día en su versión *personal* en multitud de hogares y el ciudadano medio va conociendo en creciente proporción, además de su existencia, su modo de uso y buena parte de sus posibilidades.

Pero dejando de lado esta verdadera revolución social, existen otros conceptos procedentes del desarrollo tecnológico que han superado las barreras impuestas por las industrias y centros de investigación, incorporándose en cierta medida al lenguaje coloquial. Es llamativo cómo entre éstas destaca el concepto **robot**. Aun sin tener datos reales, no parece muy aventurado suponer que de preguntar al ciudadano medio sobre qué es un robot industrial, éste demostraría tener, cuanto menos, una idea aproximada de su aspecto y utilidad.

Esta familiaridad con la que nuestra sociedad trata al robot es más llamativa cuando se compara con el amplio desconocimiento que se puede tener de otras máquinas o aparatos, aun siendo de mayor antigüedad o utilidad, como por ejemplo sería el osciloscopio o los parientes cercanos de los robots: las máquinas de control numérico.

Posiblemente una de las causas principales que haya dado popularidad al robot sea su mitificación, propiciada o amplificada por la literatura y el cine de ciencia ficción. Si bien, salvo escasas excepciones, los robots de novelas y películas tienen un nulo parecido con el robot industrial, su frecuente presencia en estos medios ha permitido que el término nos sea familiar, originando que le abramos la

puerta de nuestra cotidianeidad. Después, ha sido suficiente con que en algunas ocasiones hayamos visto un robot industrial real, por ejemplo en una noticia en televisión o prensa, para que hayamos dejado de lado al robot mito y hayamos aceptado, como una máquina más de nuestro entorno, a esa especie de brazo mecánico animado que con rapidez y precisión suelda carrocerías de vehículos o inserta circuitos integrados en placas electrónicas.

El mito ha rodeado y rodea al robot, a pesar de las más de 600.000 unidades instaladas en el mundo a principios de 1995. Sus orígenes de ficción, su controvertido impacto social, su aparente autonomía y notorio contenido tecnológico origina que, a pesar de su popularidad, siga siendo admirado y en ocasiones temido.

Este libro reúne los conocimientos necesarios para conocer a fondo qué es, cómo funciona y para qué sirve un robot. No es en general un libro para el gran público, pues en algunos capítulos se presentan conocimientos avanzados en matemáticas, electrónica, control, mecánica e informática, pero con las limitaciones impuestas en estos capítulos, puede ser leído y comprendido por cualquier persona con alguna formación técnica.

Tras su lectura es de esperar la desmitificación del robot, quedando a la altura de cualquier otra máquina compleja, como otras muchas que nos rodean (automóvil, televisión, etc.).Tras su estudio, el robot será la síntesis de un importante bagaje de conocimientos cientifico-técnicos, adecuadamente conjuntados para dar como resultado un dispositivo destinado a mejorar la producción y la calidad de vida.

1.1. ANTECEDENTES HISTÓRICOS

A lo largo de toda la historia, el hombre se ha sentido fascinado por máquinas y dispositivos capaces de imitar las funciones y los movimientos de los seres vivos. Los griegos tenían una palabra específica para denominar a estas máquinas: *automatos*. De esta palabra deriva la actual **autómata**: máquina que imita la figura y movimientos de un ser animado. Los mecanismos animados de Herón de Alejandría (85 d.C.) se movían a través de dispositivos hidráulicos, poleas y palancas y tenían fines eminentemente lúdicos.

La cultura árabe (siglos VIII a XV) heredó y difundió los conocimientos griegos, utilizándolos no sólo para realizar mecanismos destinados a la diversión, sino que les dio una aplicación práctica, introduciéndolos en la vida cotidiana de la realeza. Ejemplo de éstos son diversos sistemas dispensadores automáticos de agua para beber o lavarse. También de ese período son otros automátas, de los que hasta la actualidad no han llegado más que referencias no suficientemente documentadas, como el *Hombre de hierro* de Alberto Magno (1204-1282) o la *Cabeza parlante* de Roger Bacon (1214 - 1294). Otro ejemplo relevante de aquella época fue *el Gallo de Estrasburgo* (1352). Éste, que es el autómata más antiguo que se conserva en la actualidad, formaba parte del reloj de la torre de la catedral de Estrasburgo y al dar las horas movía las alas y el pico.

Durante los siglos XV y XVI alguno de los más relevantes representantes del renacimiento se interesan también por los ingenios descritos y desarrollados por los griegos. Es conocido el *León mecánico* construido por Leonardo Da Vinci (1452 -1519) para el rey Luis XII de Francia, que se abría el pecho con su garra y mostraba el escudo de armas del rey. En España es conocido el *Hombre de palo*,

construido por Juanelo Turriano en siglo XVI para el emperador Carlos V. Este autómata con forma de monje, andaba y movía la cabeza, ojos, boca y brazos.

Durante los siglos XVII y XVIII se crearon ingenios mecánicos que tenían alguna de las características de los robots actuales. Estos dispositivos fueron creados en su mayoría por artesanos del gremio de la relojería. Su misión principal era la de entretener a las gentes de la corte y servir de atracción en las ferias. Estos autómatas representaban figuras humanas, animales o pueblos enteros. Son destacables entre otros el pato de Vaucanson y los muñecos de la familia Droz y de Mailladert.

Jacques Vaucanson (1709-1782), autor del primer telar mecánico, construyó varios muñecos animados, entre los que destaca un flautista capaz de tocar varias melodías y un pato (1738) capaz de graznar, beber, comer, digerir y evacuar la comida. El relojero suizo Pierre Jaquet Droz (1721-1790) y sus hijos Henri-Louis y Jaquet construyeron diversos muñecos capaces de escribir (1770), dibujar (1772) y tocar diversas melodías en un órgano (1773). Estos aún se conservan en el museo de Arte e Historia de Neuchâstel, Suiza. Contemporáneo de los relojeros franceses y suizos fue Henry Maillardet, quien construyó, entre otros, una muñeca capaz de dibujar y que aún se conserva en Filadelfia.

A finales del siglo XVIII y principios del XIX se desarrollaron algunas ingeniosas invenciones mecánicas, utilizadas fundamentalmente en la industria textil, entre las que destacan la hiladora giratoria de Hargreaves (1770), la hiladora mecánica de Crompton (1779), el telar mecánico de Cartwright (1785) y el telar de Jacquard (1801). Este último utilizaba una cinta de papel perforada como un programa para las acciones de la máquina. Es a partir de este momento cuando se empiezan a utilizar dispositivos automáticos en la producción, dando paso a la automatización industrial.

Tabla 1.1. Autómatas famosos

Año	Autor	Autómata
1352	Desconocido	Gallo de la catedral de Estrasburgo
1499	L. Da Vinci	León mecánico
1525	J. Turriano	Hombre de palo
1738	J de Vaucanson	Flautista, tamborilero, pato, muñecas mecánicas de tamaño humano
1769	W. Von Kempelen	Jugador de ajedrez
1770	Familia Droz	Escriba, organista, dibujante
1805	H. Maillardet	Muñeca mecánica capaz de dibujar

La palabra **robot** fue usada por primera vez en el año 1921, cuando el escritor checo Karel Capek (1890-1938) estrena en el teatro nacional de Praga su obra *Rossum's Universal Robot* (*R.U.R.*). [CAPEK-35][CAPEK-66]. Su origen es la palabra eslava *robota*, que se refiere al trabajo realizado de manera forzada. Los robots de R.U.R. eran máquinas androides fabricadas a partir de la "formula" obtenida por un brillante científico llamado Rossum. Estos robots servían a sus jefes humanos desarrollando todos los trabajos físicos, hasta que finalmente se rebelan contra sus dueños, destruyendo toda la vida humana, a excepción de uno de sus creadores, con la frustrada esperanza de que les enseñe a reproducirse.

El término posiblemente hubiera caído en desuso si no hubiese sido por los escritores del género literario de la ciencia ficción, algunos de los cuales retomaron la palabra, e incluso el mensaje de la obra de Capek: la dominación de la especie humana por seres hechos a su propia imagen. Así, en 1926, Thea von Harbou escribe *Metrópolis*, novela posteriormente llevada al cine por su marido Fritz Lang, en donde la masa obrera de una sociedad superindustrializada es manipulada por un lider androide llamado María.

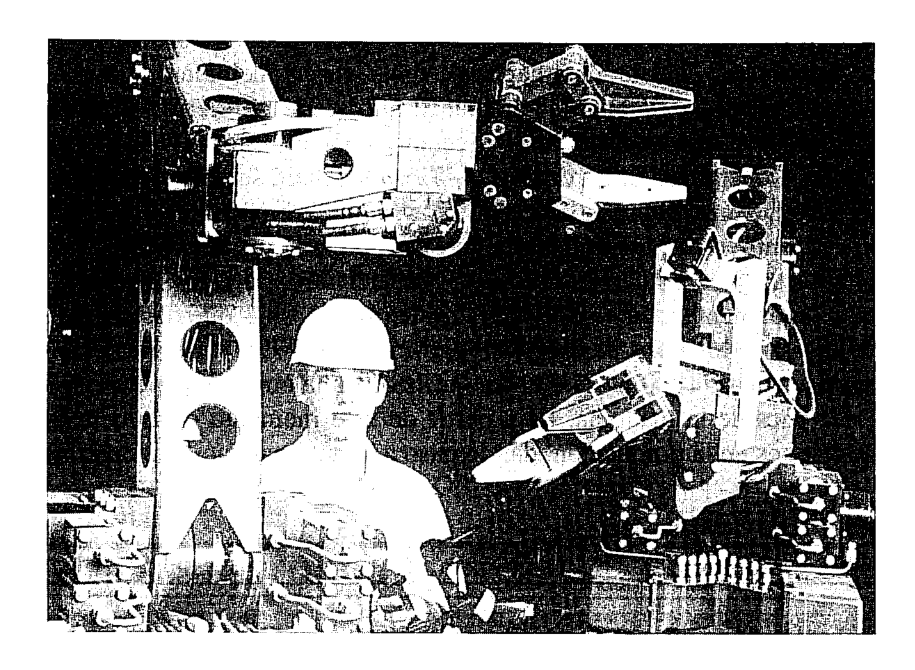


Figura 1.1. Sistema de telemanipulación bilateral.

Pero sin duda alguna, fue el escritor americano de origen ruso Isaac Asimov (1920-1992) el máximo impulsor de la palabra robot. En octubre de 1945 publicó en la revista *Galaxy Science Fiction* una historia en la que por primera vez enunció sus *tres leyes de la robótica*[1].

1. Un robot no puede perjudicar a un ser humano, ni con su inacción permitir que un ser humano sufra daño.
2. Un robot ha de obedecer las órdenes recibidas de un ser humano, excepto si tales órdenes entran en conflicto con la primera ley.
3. Un robot debe proteger su propia existencia mientras tal protección no entre en conflicto con la primera o segunda ley.

[1] En la novela *Robots e Imperio*, publicada en 1985, Asimov incorporó una cuarta ley, conocida como ley cero: un robot no puede lastimar a la humanidad o, por falta de acción, permitir que la humanidad sufra daño. Esta ley, de mayor prioridad que la primera (que debe ser modificada en tal sentido), antepone el bien comunitario al individual.

Se le atribuye a Asimov la creación del término *robotics* (robótica) y sin lugar a duda, desde su obra literaria, ha contribuido decisivamente a la divulgación y difusión de la robótica.

1.2. ORIGEN Y DESARROLLO DE LA ROBÓTICA

El robot como máquina lleva un desarrollo independiente del término robot. Tras los primeros autómatas descritos en el apartado anterior, casi todos de aspecto humano, los progenitores más directos de los robots fueron los **telemanipuladores** (Figura 1.1). En 1948 R.C. Goertz del Argonne National Laboratory desarrolló, con el objetivo de manipular elementos radioactivos sin riesgo para el operador, el primer telemanipulador. Éste consistía en un dispositivo mecánico maestro-esclavo. El manipulador maestro, situado en la zona segura, era movido directamente por el operador, mientras que el esclavo, situado en contacto con los elementos radioactivos y unido mecánicamente al maestro, reproducía fielmente los movimientos de éste. El operador además de poder observar a través de un grueso cristal el resultado de sus acciones, sentía a través del dispositivo maestro, las fuerzas que el esclavo ejercía sobre el entorno.

Años más tarde, en 1954, Goertz hizo uso de la tecnología electrónica y del servocontrol sustituyendo la transmisión mecánica por otra eléctrica y desarrollando así el primer telemanipulador con servocontrol bilateral. Otro de los pioneros de la telemanipulación fue Ralph Mosher, ingeniero de la General Electric que en 1958 desarrolló un dispositivo denominado Handy-Man, consistente en 2 brazos mecánicos teleoperados mediante un maestro del tipo denominado exoesqueleto. Junto a la industria nuclear, a lo largo de los años sesenta la industria submarina comenzó a interesarse por el uso de los telemanipuladores. A este interés se sumó la industria espacial en los años setenta.

La evolución de los telemanipuladores a lo largo de los últimos años no ha sido tan espectacular como la de los robots. Recluidos en un mercado selecto y limitado (industria nuclear, militar, espacial, etc.) son en general desconocidos y comparativamente poco atendidos por los investigadores y usuarios de robots. Por su propia concepción, un telemanipulador precisa del mando continuo de un operador, y salvo por las aportaciones incorporadas con el concepto del control supervisado y la mejora de la telepresencia promovida hoy en día por la realidad virtual, sus capacidades no han variado mucho respecto a las de sus origenes.

La sustitución del operador por un programa de ordenador que controlase los movimientos del manipulador dio paso al concepto de robot.

La primera patente de un dispositivo robótico fue solicitada en marzo de 1954 por el inventor británico C.W. Kenward. Dicha patente fue emitida en el Reino Unido en 1957 con el número 781465. Sin embargo, fue George C. Devol, ingeniero norteamericano, inventor y autor de varias patentes, el que estableció las bases del robot industrial moderno. En 1954 Devol concibió la idea de un dispositivo de *transferencia de artículos programada* que se patentó en Estados Unidos en 1961 con el número 2988237. En 1956 pone esta idea en conocimiento de Joseph F. Engelberger (1925-), ávido lector de Asimov y director de ingeniería de la división aeroespacial de la empresa Manning Maxwell y Moore en Stanford, Conneticut. Juntos, Devol y Engelberger comienzan a trabajar en la utilización industrial de sus máquinas, fundando la Consolidated Controls Corporation, que más tarde se con-

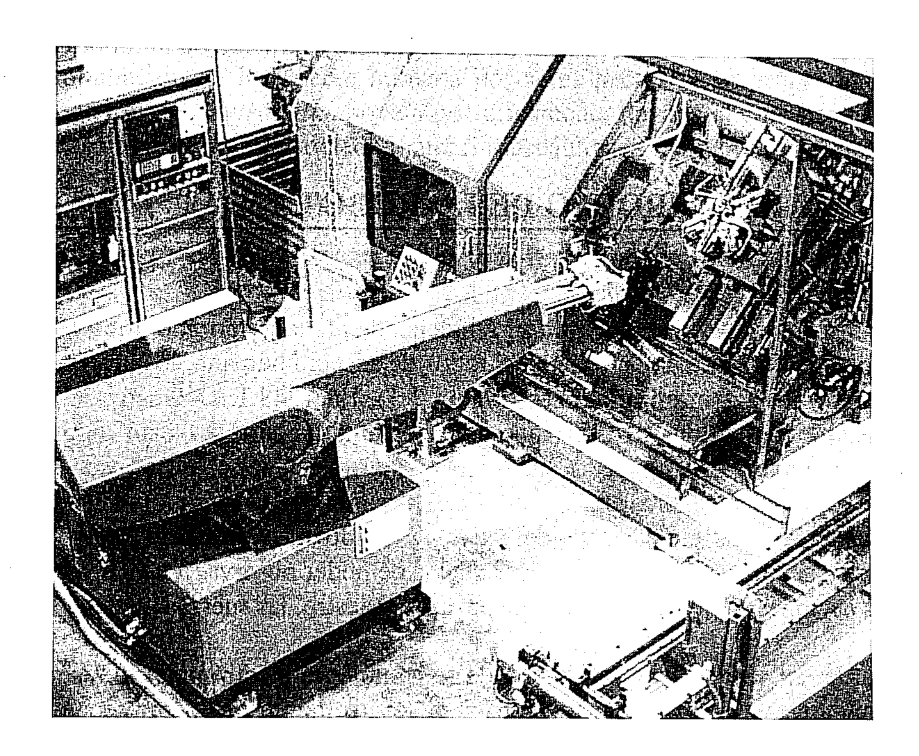


Figura 1.2. Robot Unimate 2000. Uno de los primeros modelos comerciales de Unimation. (Cortesía de Staübli.)

vierte en Unimation (Universal Automation), e instalando su primera máquina Unimate (1960) similar a la de la Figura 1.2 en la fábrica de General Motors de Trenton, Nueva Jersey, en una aplicación de fundición por inyección. Otras grandes empresas, como la AMF, emprendieron la construcción de máquinas similares (Versatran, 1963), que más tarde por motivos comerciales se denominaron con el término robot procedente de la ciencia ficción y a pesar de no contar con la apariencia humana de aquellos.

En 1968 J.F. Engelberger visitó Japón y poco más tarde se firmaron acuerdos con Kawasaki para la construcción de robots tipo Unimate. El crecimiento de la robótica en Japón aventajó en breve a los Estados Unidos gracias a Nissan, que formó la primera asociación robótica del mundo, la *Asociación de Robótica Industrial de Japón* (JIRA) en 1972. Dos años más tarde se formó el *Instituto de Robótica de América* (RIA), que en 1984 cambió su nombre por el de *Asociación de Industrias Robóticas*, manteniendo las mismas siglas (RIA).

Por su parte Europa tuvo un despertar más tardío. En 1973 la firma sueca ASEA construyó el primer robot con accionamiento totalmente eléctrico, el robot IRb6 (Figura 1.3), seguido un año más tarde del IRb60. En 1980 se fundó la *Federación Internacional de Robótica* con sede en Estocolmo, Suecia.

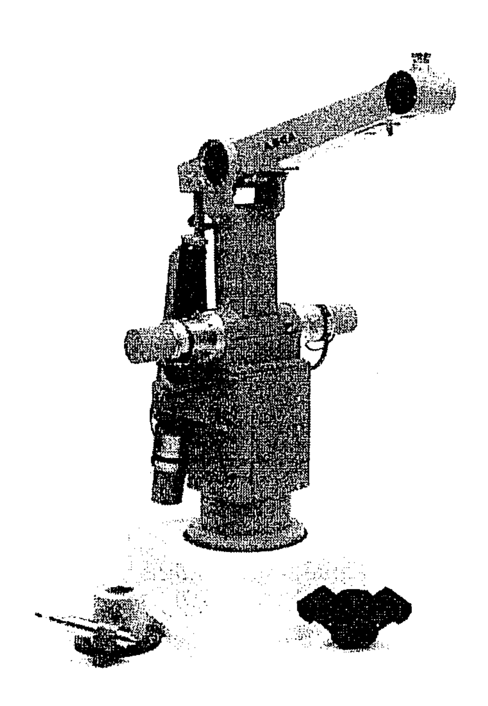


Figura 1.3. Robot IRb6 de la firma sueca ASEA. (Cortesía de ABB Robótica.)

La configuración de los primeros robots respondía a las denominadas configuraciones esférica y antropomórfica, de uso especialmente válido para la manipulación. En 1982, el profesor Makino de la Universidad Yamanashi de Japón, desarrolla en concepto de robot SCARA (*Selective Compliance Assembly Robot Arm*) que busca un robot con un número reducido de grados de libertad (3 o 4), un coste limitado y una configuración orientada al ensamblado de piezas.

La evolución de los robots industriales desde sus primeros balbuceos ha sido vertiginosa. En poco más de 30 años las investigaciones y desarrollos sobre robótica industrial han permitido que los robots tomen posiciones en casi todas las áreas productivas y tipos de industria. En pequeñas o grandes fábricas, los robots pueden sustituir al hombre en aquellas tareas repetitivas y hostiles, adaptándose inmediatamente a los cambios de producción solicitados por la demanda variable. En la Figura 1.4 se muestra uno de los últimos modelos comerciales de robot industrial.

Los futuros desarrollos de la robótica apuntan a aumentar su movilidad, destreza y autonomía de sus acciones. La mayor parte de los robots actuales son con base estática, y se utilizan en aplicaciones industriales tales como ensamblado, soldadura, alimentación de máquinas herramientas, etc. Sin embargo, existen otro tipo de aplicaciones que han hecho evolucionar en gran medida tanto la concepción de los robots como su propia morfología. Entre estos robots dedicados a aplicaciones no industriales destacan los robots espaciales (brazos para lanzamiento y recuperación de satélites, vehículos de exploración lunar, robots para construcción y mantenimiento de hardware en el espacio); robots para aplicaciones submarinas y subterráneas (exploración submarina, instalación y mantenimiento de

cables telefónicos submarinos, limpieza e inspección de tuberías y drenajes subterráneos, inspección de sistemas de refrigeración de centrales nucleares); robots militares (desactivación de bombas, robots centinelas experimentales dedicados a patrullar áreas críticas) ; robots móviles industriales (robots bomberos para patrullar fábricas, robots bibliotecarios, robots andantes con piernas); aplicaciones médicas (prótesis robotizadas, sistemas de ayuda a discapacitados); aplicaciones agrícolas (sembrado y recogida de cosechas, robot para esquilar ovejas); y un largo etcétera. Parece como si la robótica y los telemanipuladores, volviendo los ojos hacia sus orígenes y con la experiencia acumulada tras 30 años, reemprendiesen, esta vez juntos, el camino que un día tomaron de manera divergente.

Puede que en un futuro los robots industriales se parezcan a aquellos robots de las novelas de Asimov, Capek o Harbou, que dieron nombre a los actuales herederos del Unimate.

1.3. DEFINICIÓN Y CLASIFICACIÓN DEL ROBOT

La robótica está experimentando en la actualidad una notoria ampliación de sus campos de actuación. Saturadas en cierta medida las aplicaciones del robot en talleres y líneas de fabricación, los robots están tomando posiciones en aplicaciones fuera del taller, caracterizadas fundamentalmente por la falta de

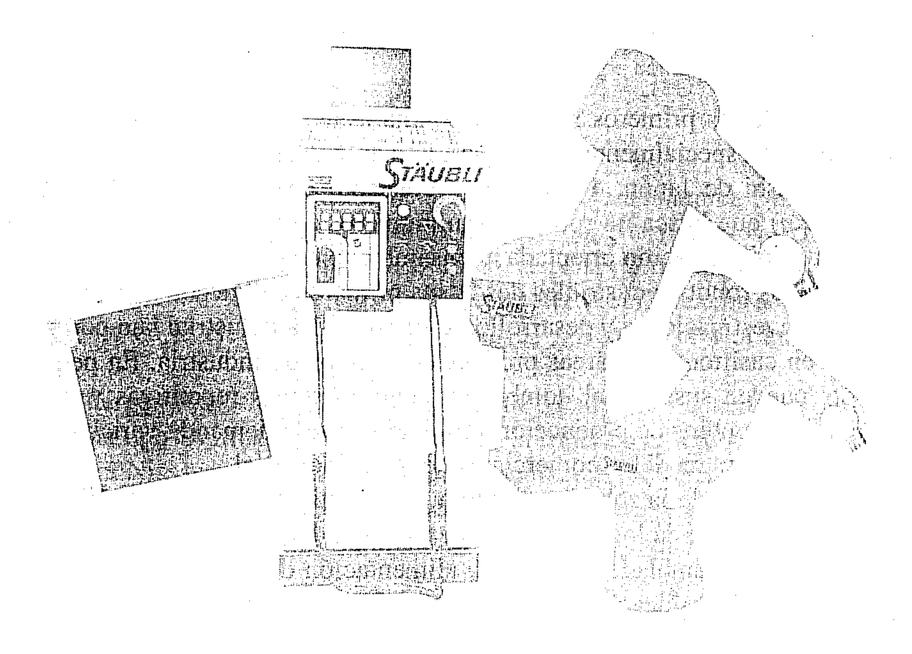


Figura 1.4. Robots Staübli tipo RX pertenecientes a la última generación. (Cortesía de Staübli.)

estandarización del producto y del proceso y la no estructuración del entorno.

Buena parte de las definiciones y clasificaciones de robots existentes responde al robot ampliamente utilizado hasta la fecha, destinado a la fabricación flexible de series medias y que se conoce como **robot industrial** o robot de producción. Frente a éstos, los robots especiales, también denominados robots de servicio, están aún en un estado de desarrollo incipiente, aunque es previsible un considerable desarrollo de los mismos. Las definiciones y generaciones aquí presentadas responden fundamentalmente al robot industrial.

1.3.1. Definición de robot industrial

Existen ciertas dificultades a la hora de establecer una definición formal de lo que es un robot industrial. La primera de ellas surge de la diferencia conceptual entre el mercado japonés y el euro-americano de lo que es un robot y lo que es un manipulador. Así, mientras que para los japoneses un robot industrial es cualquier dispositivo mecánico dotado de articulaciones móviles destinado a la manipulación (ver la Figura 1.5), el mercado occidental es más restrictivo, exigiendo una mayor complejidad, sobre todo en lo relativo al control.

En segundo lugar, y centrándose ya en el concepto occidental, aunque existe una idea común acerca de lo que es un robot industrial, no es fácil ponerse de acuerdo a la hora de establecer una definición formal. Además, la evolución de la robótica ha ido obligando a diferentes actualizaciones de su definición.

La definición más comúnmente aceptada posiblemente sea la de la *Asociación de Industrias Robóticas* (RIA), según la cual:

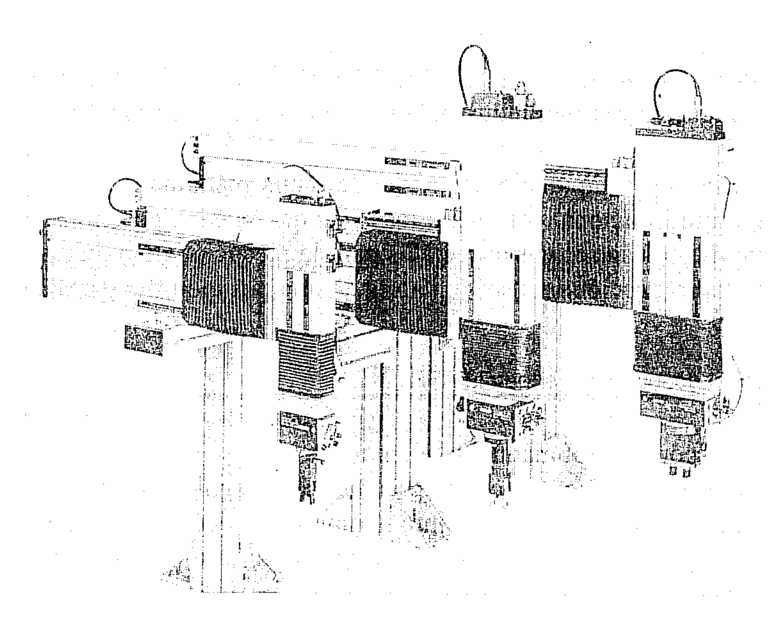


Figura 1.5. Manipuladores de accionamiento neumático. (Cortesía de Bosch.)

- Un robot industrial es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas.

Esta definición, ligeramente modificada, ha sido adoptada por la *Organización Internacional de Estándares* (ISO) que define al robot industrial como:

- Manipulador multifuncional reprogramable con varios grados de libertad, capaz de manipular materias, piezas, herramientas o dispositivos especiales según trayectorias variables programadas para realizar tareas diversas.

Se incluye en esta definición la necesidad de que el robot tenga varios grados de libertad. Una definición más completa es la establecida por la *Asociación Francesa de Normalización* (AFNOR) que define primero el manipulador y, basándose en dicha definición, el robot:

- Manipulador: mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico.

- Robot: manipulador automático servocontrolado, reprogramable, polivalente, capaz de posicionar y orientar piezas, útiles o dispositivos especiales, siguiendo trayectorias variables reprogramables, para la ejecución de tareas variadas. Normalmente tiene la forma de uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su material.

Por último, la *Federación Internacional de Robótica* (IFR) en su informe técnico ISO/TR 83737 (septiembre 1988) distingue entre robot industrial de manipulación y otros robots:

- Por robot industrial de manipulación se entiende a una máquina de manipulación automática, reprogramable y multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento.

Tabla 1.2. Clasificación de los robots según la AFRI

Tipo A	Manipulador con control manual o telemando.
Tipo B	Manipulador automático con ciclos preajustados; regulación mediante fines de carrera o topes; control por PLC; accionamiento neumático, eléctrico o hidráulico.
Tipo C	Robot programable con trayectoria continua o punto a punto. Carece de conocimientos sobre su entorno.
Tipo D	Robot capaz de adquirir datos de su entorno, readaptando su tarea en función de éstos.

Tabla 1.3. Clasificación de los robots industriales en generaciones

1ª Generación	Repite la tarea programada secuencialmente. No toma en cuenta las posibles alteraciones de su entorno.
2ª Generación	Adquiere información limitada de su entorno y actúa en consecuencia. Puede localizar, clasificar (visión) y detectar esfuerzos y adaptar sus movimientos en consecuencia.
3ª Generación	Su programación se realiza mediante el empleo de un lenguaje natural. Posee capacidad para la planificación automática de tareas.

En esta definición se debe entender que la reprogramabilidad y multifunción se consigue sin modificaciones físicas del robot.

Común a todas las definiciones anteriores es la aceptación del robot industrial como un brazo mecánico con capacidad de manipulación y que incorpora un control más o menos complejo. Un sistema robotizado, en cambio, es un concepto más amplio. Engloba todos aquellos dispositivos que realizan tareas de forma automática en sustitución de un ser humano y que pueden incorporar o no a uno varios robots, siendo esto último lo más frecuente.

1.3.2. Clasificación del robot industrial

La IFR distingue entre cuatro tipos de robots:

- Robot secuencial.
- Robot de trayectoria controlable.
- Robot adaptativo.
- Robot telemanipulado.

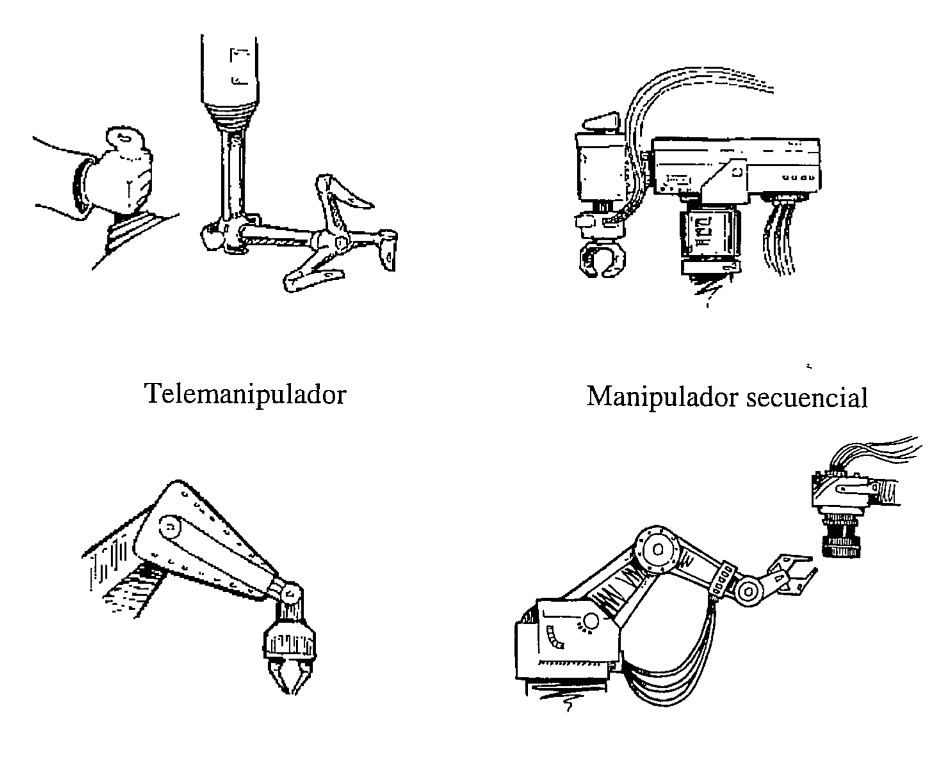
Esta clasificación coincide en gran medida con la establecida por la *Asociación Francesa de Robótica Industrial* (AFRI)(Tabla 1.2). (Más simple y específica es la clasificación de los robots según generaciones recogida en la Tabla 1.3.)

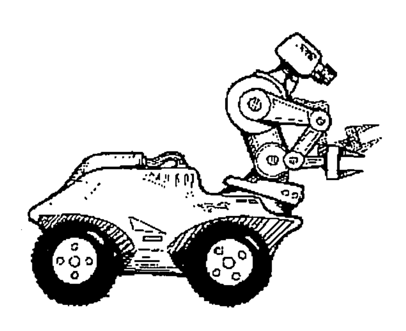
1.3.3. Robots de servicio y teleoperados

En cuanto a los **robots de servicio**, se pueden definir como:

- Dispositivos electromecánicos móviles o estacionarios, dotados normalmente de uno o varios brazos mecánicos independientes, controlados por un programa de ordenador y que realizan tareas no industriales de servicio [NOSTRAND-90].

En esta definición entrarían entre otros los robots dedicados a cuidados médicos, educación, domésticos, uso en oficinas, intervención en ambientes peligrosos, aplicaciones espaciales, aplicaciones submarinas y agricultura. Sin embargo, esta definición de robots de servicio excluye los telemanipuladores, pues éstos no se mueven mediante el control de un programa de ordenador, sino que están controlados directamente por el operador humano.





Robot de actuación en exteriores

Figura 1.6. Algunos tipos de robots actualmente existentes (I).

Los **robots teleoperados** son definidos por la NASA (1978) como:

Dispositivos robóticos con brazos manipuladores y sensores y cierto grado de movilidad, controlados remotamente por un operador humano de manera directa o a través de un ordenador.

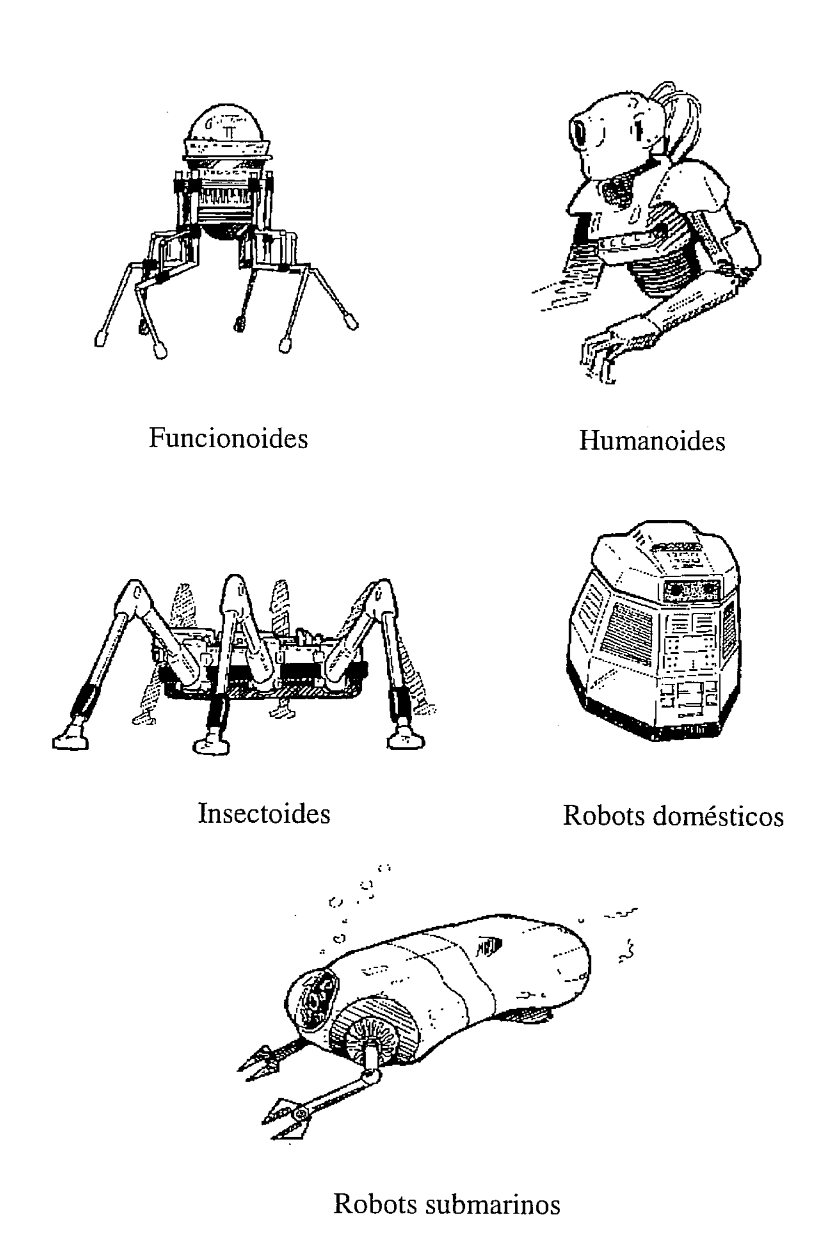


Figura 1.7. Algunos tipos de robos actualmente existentes (II).

Tabla 1.4. Clasificación de los robots según T.M. Knasel

Generación	Nombre	Tipo de control	Grado de movilidad	Usos más frecuentes
1ª (1982)	Pick & place	Fines de carrera, aprendizaje	Niguno	Manipulación, servicio de máquinas
2ª (1984)	Servo	Servocontrol, trayectoria continua, progr. codicional	Desplazamiento por vía	Soldadura, pintura
3ª (1989)	Ensamblado	Servos de precisión, visión, tacto, prog. off-line	AGV Guiado por vía	Ensamblado Desbarbado
4ª (2000)	Móvil	Sensores inteligentes	Patas Ruedas	Construcción Mantenimiento
5ª (2010)	Especiales	Controlados con técnicas de IA	Andante Saltarín	Uso militar Uso espacial

Por último y con el fin de dar una visión del posible futuro, se incluye en la Tabla 1.4 la clasificación propuesta por T.M. Knasel [KNASEL-86]. La Figura 1.6 y la Figura 1.7 recrean buena parte de los diversos tipos de robots que se pueden encontrar hoy en día. Todos los robots representados existen en la actualidad, aunque los casos más futuristas están en estado de desarrollo en los centros de investigación de robótica.

1.4. BIBLIOGRAFÍA

[ASIMOV-85] I. Asimov, K.A. Frenkel, *Robots. Máquinas a imagen y semejanza del hombre*, Plaza y Janés, 1985.

[BYRON-94] Byron Press Multimedia, *Isaac Asimov's The Ultimate Robot*, CD-ROM media, Microsoft Corp., 1994.

[CAPEK-35] K. Capek, *R.U.R Rossum's Universal Robots*, F.R. Borový, Praga, 1935 (Edición en Checo).

[CAPEK-66] K. Capek, *R.U.R. Robots Universales Rossum*, Alianza Editorial, Libro de bolsillo, Madrid, 1966.

[KNASEL-86] T.M. Kansel, "Mobile Robotics. State of the art review", en: *Robotics 2*, North-Holland, 1986.

[NOSTRAND-90] J.W. Nostrand y E.L. Sampson, "Robots in Service Industry", en: *Concise International Encyclopledia of Robotics*, Editor R.C. Dorf, John Wiley & Sons, 1990.

[ROSHEIM-94] M.E. Rosheim, *Robot Evolution. The Development of Anthrobotics*, John Wiley & Sons, 1994.

Capítulo 2

MORFOLOGÍA DEL ROBOT

Cada uno cuenta de la feria como le va en ella.
Refranero popular

Un robot está formado por los siguientes elementos: estructura mecánica, transmisiones, sistema de accionamiento, sistema sensorial, sistema de control y elementos terminales.

En este capítulo se examinan estos elementos constitutivos de un robot. Se estudia primero la estructura mecánica, haciendo referencia a los distintos tipos de articulaciones posibles entre dos eslabones consecutivos y a las configuraciones existentes. Seguidamente se analizan los sistemas de transmisión y reducción necesarios para transmitir a cada una de las articulaciones el movimiento generado por los actuadores, que como elementos con capacidad para desplazar la estructura mecánica, son examinados en el siguiente apartado.

Se repasan después los denominados sensores internos, necesarios para el control de los movimientos del robot. Finalmente, el último apartado se ocupa de los elementos terminales (pinzas, herramientas, dispositivos de sujeción, etc.), que situados generalmente en el extremo del robot, sirven para que éste pueda interaccionar con el mundo exterior realizando las operaciones que le han sido asignadas.

Aunque los elementos empleados en los robots no son exclusivos de éstos (máquinas herramientas y otras muchas máquinas emplean tecnologías semejantes), las altas prestaciones que se exigen a los robots han motivado que en ellos se empleen elementos con características específicas.

La Figura 2.1 muestra la estructura mecánica de un robot, sobre la que se ha indicado alguno de los elementos que se van a estudiar en este capítulo, examinando las necesidades particulares de cada uno y comparando las diversas alternativas existentes. Se señalarán además cuáles son las tendencias actuales y los desarrollos que se prevén para el futuro.

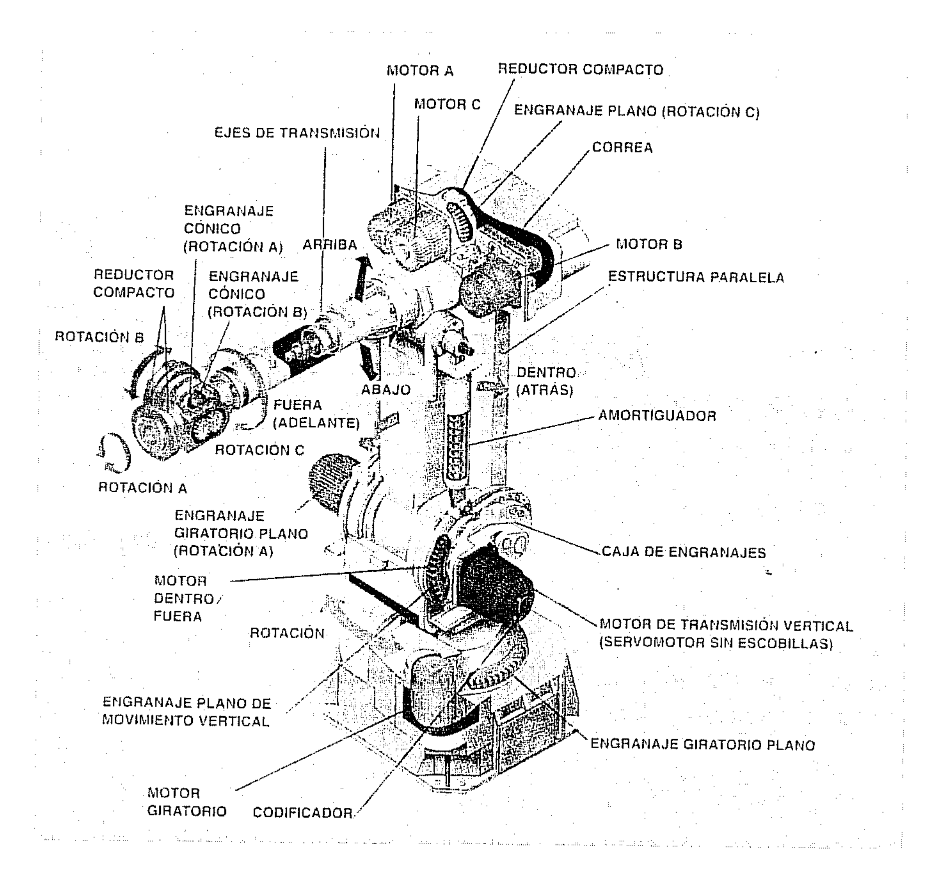


Figura 2.1. Ejemplo de estructura mécanica y elementos constitutivos de un robot. (Cortesía de INSER Robótica.)

2.1. ESTRUCTURA MECÁNICA DE UN ROBOT

Mecánicamente, un robot está formado por una serie de elementos o eslabones unidos mediante articulaciones que permiten un movimiento relativo entre cada dos eslabones consecutivos. La constitución física de la mayor parte de los robots industriales guarda cierta similitud con la anatomía del brazo humano, por lo que en ocasiones, para hacer referencia a los distintos elementos que componen el robot, se usan términos como cuerpo, brazo, codo y muñeca.

El movimiento de cada articulación puede ser de desplazamiento, de giro, o de una combinación de ambos. De este modo son posibles los seis tipos diferentes de articulaciones que se muestran en la Figura 2.2, aunque, en la práctica, en los robots sólo se emplean la de rotación y la prismática.

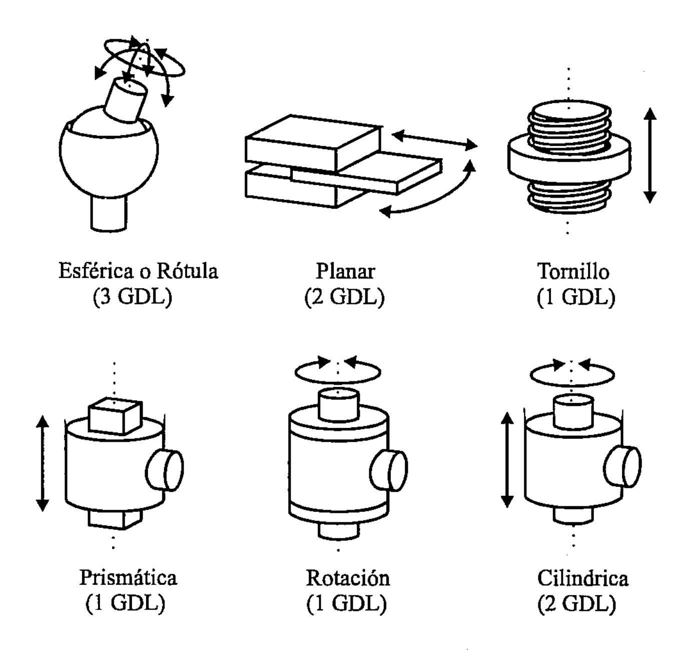


Figura 2.2. Distintos tipos de articulaciones para robots.

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se denomina **grado de libertad** (GDL). En la Figura 2.2 se indica el número de GDL de cada tipo de articulación. El número de grados de libertad del robot viene dado por la suma de los grados de libertad de las articulaciones que lo componen. Puesto que, como se ha indicado, las articulaciones empleadas son únicamente las de rotación y prismática con un solo GDL cada una, el número de GDL del robot suele coincidir con el número de articulaciones de que se compone.

El empleo de diferentes combinaciones de articulaciones en un robot, da lugar a diferentes configuraciones, con características a tener en cuenta tanto en el diseño y construcción del robot como en su aplicación. Las combinaciones más frecuentes son las representadas en la Figura 2.3 donde se atiende únicamente a las tres primeras articulaciones del robot, que son las más importantes a la hora de posicionar su extremo en un punto del espacio.

Puesto que para posicionar y orientar un cuerpo de cualquier manera en el espacio son necesarios seis parámetros, tres para definir la posición y tres para la orientación, si se pretende que un robot posicione y oriente su extremo (y con él la pieza o herramienta manipulada) de cualquier modo en el espacio, se precisarán al menos seis GDL.

En la Figura 2.4 se muestran los seis GDL con que está dotado el robot ARC Mate 120/S-12 de Fanuc, así como sus articulaciones y eslabones.

En la práctica, a pesar de ser necesarios los seis GDL comentados para tener total libertad en el posicionado y orientación del extremo del robot, muchos robots industriales cuentan con sólo cuatro o cinco GDL, por ser estos suficientes para llevar a cabo las tareas que se les encomiendan.

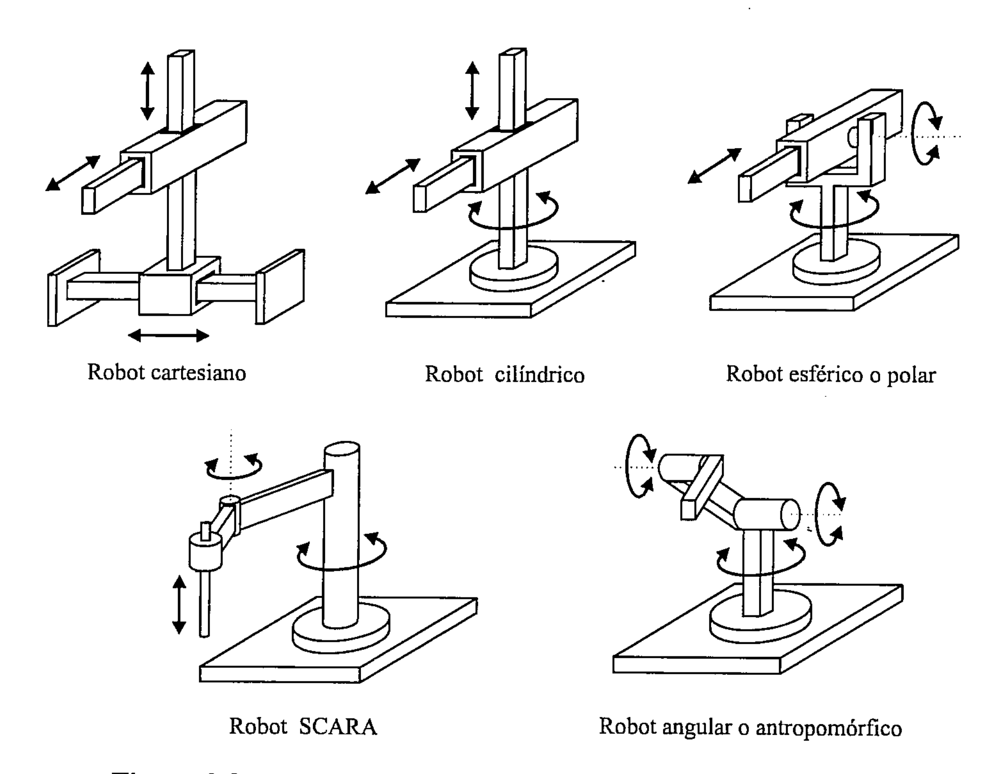


Figura 2.3. Configuraciones más frecuentes en robots industriales.

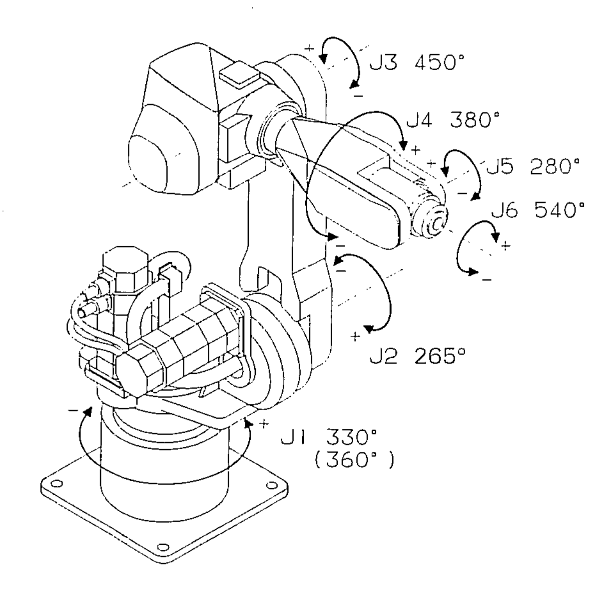


Figura 2.4. Grados de libertad del robot ARC Mate 120/S-12. (Cortesía de FANUC.)

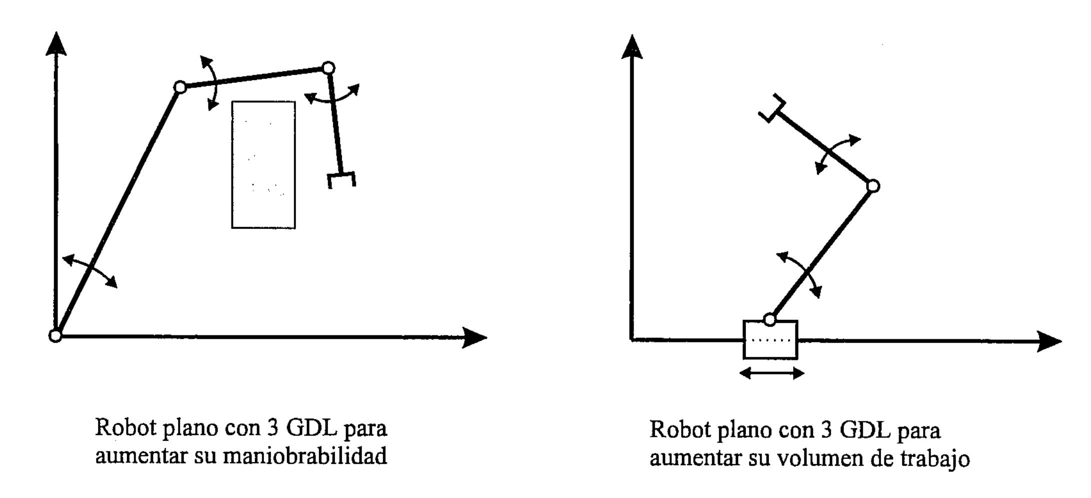


Figura 2.5. Robots planares redundantes.

Existen también casos opuestos, en los que se precisan más de seis GDL para que el robot pueda tener acceso a todos los puntos de su entorno. Así, si se trabaja en un entorno con obstáculos, el dotar al robot de grados de libertad adicionales le permitirá acceder a posiciones y orientaciones de su extremo a las que, como consecuencia de los obstáculos, no hubiera llegado con seis GDL. Otra situación frecuente es la de dotar al robot de un GDL adicional que le permita desplazarse a lo largo de un carril, aumentando así el volumen del espacio al que puede acceder. Cuando el número de grados de libertad del robot es mayor que los necesarios para realizar una determinada tarea se dice que el robot es redundante.

La Figura 2.5 representa las dos situaciones comentadas para el caso de robots planares a los que les bastaría con 2 GDL para posicionar su extremo en cualquier punto del plano.

2.2. TRANSMISIONES Y REDUCTORES

Las **transmisiones** son los elementos encargados de transmitir el movimiento desde los actuadores hasta las articulaciones. Se incluirán junto con las transmisiones a los **reductores**, encargados de adaptar el par y la velocidad de la salida del actuador a los valores adecuados para el movimiento de los elementos del robot.

2.2.1. Transmisiones

Dado que un robot mueve su extremo con aceleraciones elevadas, es de gran importancia reducir al máximo su momento de inercia. Del mismo modo, los pares estáticos que deben vencer los actuadores dependen directamente de la distancia de las masas al actuador. Por estos motivos se procura que los actuadores, por lo general pesados, estén lo más cerca posible de la base del robot. Esta circunstancia obliga a utilizar sistemas de transmisión que trasladen el movimiento hasta las articulaciones,

Tabla 2.1. Sistemas de transmisión para robots

Entrada-Salida	Denominación	Ventajas	Inconvenientes
Circular-Circular	Engranaje	Pares altos	Holguras
	Correa dentada	Distancia grande	-
	Cadena	Distancia grande	Ruido
	Paralelogramo	-	Giro limitado
	Cable	-	Deformabilidad
Circular-Lineal	Tornillo sin fin	Poca holgura	Rozamiento
	Cremallera	Holgura media	Rozamiento
Lineal-Circular	Paral. articulado	-	Control difícil
	Cremallera	Holgura media	Rozamiento

especialmente a las situadas en el extremo del robot. Asimismo, las transmisiones pueden ser utilizadas para convertir movimiento circular en lineal o viceversa, lo que en ocasiones puede ser necesario.

Existen actualmente en el mercado robots industriales con acoplamiento directo entre accionamiento y articulación, ventajosos, tal y como se detalla más adelante, en numerosas ocasiones. Se trata, sin embargo, de casos particulares dentro de la generalidad que en los robots industriales actuales supone la existencia de sistemas de transmisión junto con reductores para el acoplamiento entre actuadores y articulaciones.

Es de esperar que un buen sistema de transmisión cumpla una serie de características básicas: debe tener un tamaño y peso reducido, se ha de evitar que presente juegos u holguras considerables y se deben buscar transmisiones con gran rendimiento.

Aunque no existe un sistema de transmisión específico para robots, sí existen algunos usados con mayor frecuencia y que se recogen clasificados en la Tabla 2.1. La clasificación se ha realizado en base al tipo de movimiento posible en la entrada y la salida: lineal o circular. En la citada tabla también quedan reflejadas algunas ventajas e inconvenientes propios de algunos sistemas de transmisión. Entre ellas cabe destacar la holgura o juego. Es muy importante que el sistema de transmisión a utilizar no afecte al movimiento que transmite, ya sea por el rozamiento inherente a su funcionamiento o por las holguras que su desgaste pueda introducir. También hay que tener en cuenta que el sistema de transmisión sea capaz de soportar un funcionamiento continuo a un par elevado, y a ser posible entre grandes distancias.

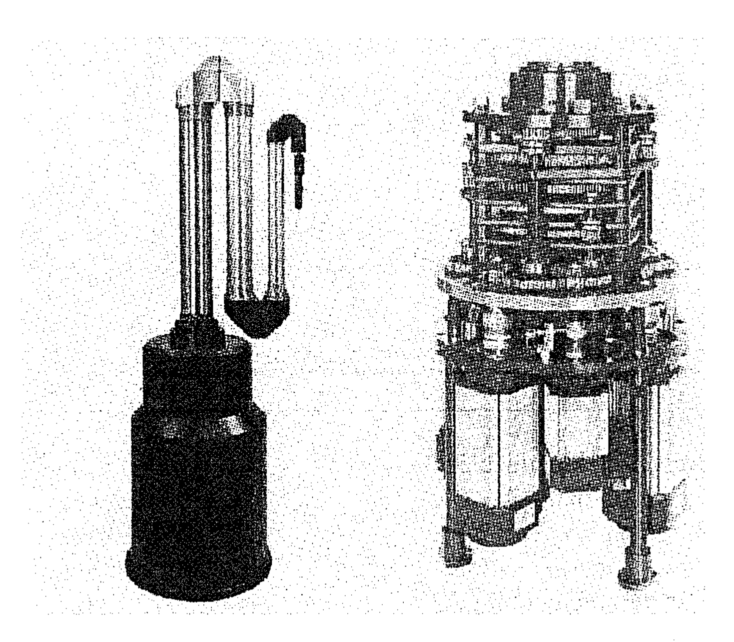


Figura 2.6. Telemanipulador TELBOT junto con la caja de transmisión por engranajes situada en su base. (Cortesía de Wälischmiller GmbH.)

Las transmisiones más habituales son aquellas que cuentan con movimiento circular tanto a la entrada como a la salida. Incluidas en éstas se hallan los engranajes, las correas dentadas y las cadenas. La Figura 2.6 muestra el brazo articulado de un telemanipulador junto con los engranajes de transmisión que transmiten el par de los motores situados en su base a los eslabones.

2.2.2. Reductores

En cuanto a los reductores, al contrario que con las transmisiones, sí que existen determinados sistemas usados de manera preferente en los robots industriales. Esto se debe a que a los reductores utilizados en robótica se les exige unas condiciones de funcionamiento muy restrictivas. La exigencia de estas características viene motivada por las altas prestaciones que se le piden al robot en cuanto a precisión y velocidad de posicionamiento. La Tabla 2.2 muestra valores típicos de los reductores para robótica actualmente empleados.

Se buscan reductores de bajo peso, reducido tamaño, bajo rozamiento y que al mismo tiempo sean capaces de realizar una reducción elevada de velocidad en un único paso. Se tiende también a minimizar su momento de inercia, de negativa influencia en el funcionamiento del motor, especialmente crítico en el caso de motores de baja inercia [FERRANDO-93].

Los reductores, por motivos de diseño, tienen una velocidad máxima de entrada admisible, que como regla general aumenta a medida que disminuye el tamaño del motor. También existe una limita-

ción en cuanto al par de salida nominal permisible (T_2) que depende del par de entrada (T_1) y de la relación de transmisión a través de la relación:

$$T_2 = \eta T_1 \frac{\omega_1}{\omega_2} \qquad [2.\ 1]$$

donde el rendimiento (η) puede llegar a ser cerca del 100 % y la relación de reducción de velocidades (ω_1 = velocidad de entrada; ω_2 = velocidad de salida) varía entre 50 y 300.

Tabla 2.2. Características de reductores para robótica

Características	Valores típicos
Relación de reducción	50 ÷ 300
Peso y tamaño	0.1 ÷ 30 kg
Momento de inercia	10^{-4} kg m^2
Velocidades de entrada máxima	6000 ÷ 7000 rpm
Par de salida nominal	5700 Nm
Par de salida máximo	7900 Nm
Juego angular	0 ÷ 2 ″
Rigidez torsional	100÷ 2000 Nm/rad
Rendimiento	85 % ÷ 98 %

Puesto que los robots trabajan en ciclos cortos que implican continuos arranques y paradas, es de gran importancia que el reductor sea capaz de soportar pares elevados puntuales. También se busca que el juego angular o *backlash* sea lo menor posible. Éste se define como el ángulo que gira el eje de salida cuando se cambia su sentido de giro sin que llegue a girar el eje de entrada. Por último, es importante que los reductores para robótica posean una alta rigidez torsional, definida como el par que hay que aplicar sobre el eje de salida para que, manteniendo bloqueado el de entrada, aquél gire un ángulo unitario.

Los reductores para robots más comúnmente usados son los de las empresas Harmonic Drive [HARMONIC-96] y Cyclo-Getriebebau [CYCLO-96]. Los primeros, denominados HDUC (Figura 2.7), se basan en una corona exterior rígida (Figura 2.8) con dentado interior (*circular spline*) y un vaso flexible (*flexspline*) con dentado exterior que engrana en el primero. El número de dientes de ambos difiere en 1 o 2. Interiormente al vaso gira un rodamiento elipsoidal (*wave generator*) que deforma el vaso, poniendo en contacto la corona exterior con la zona del vaso correspondiente al máximo diámetro de la elipse. Al girar el *wave generator* (al que se fija el eje de entrada), se obliga a que los dientes del *flexspline* (fijado al eje de salida) engranen uno a uno con los del *circular spline*, de modo que al haber una diferencia de dientes $Z = N_c - N_f$, tras una vuelta completa del *wave generator*, el *flexspline* solo habrá avanzado Z dientes. La relación de reducción conseguida será, por tanto, de Z/N_f. En concreto, se consiguen reducciones de hasta 320, con una holgura cercana a cero y capacidad de transmisión de par de 5720 Nm.

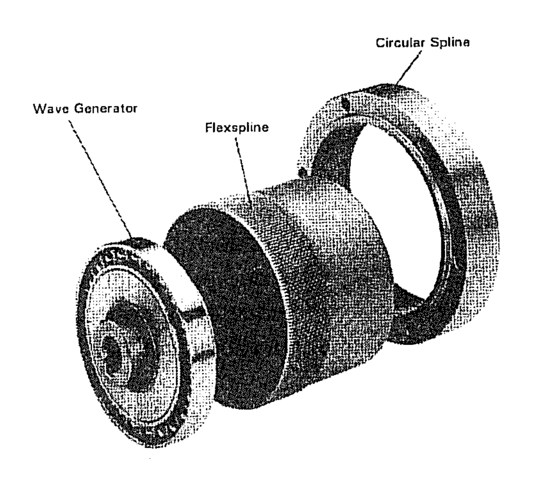


Figura 2.7. Despiece HDUC.
(Cortesía de HarmonicDrive.)

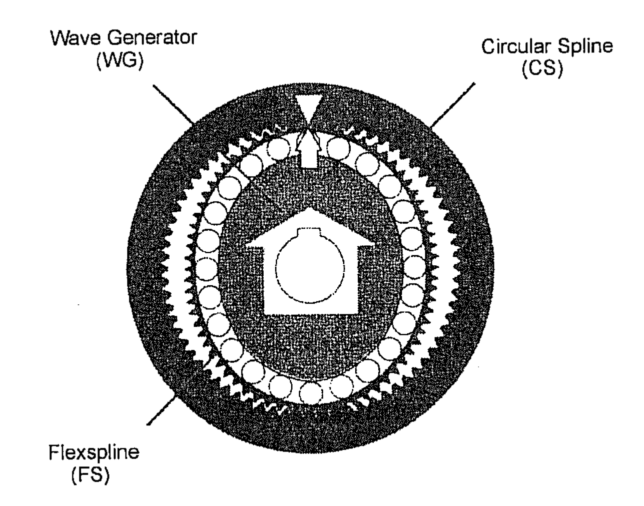


Figura 2.8. Esquema HDUC.
(Cortesía de HarmonicDrive.)

Por otra parte el sistema de reducción de los reductores CYCLO se basa en el movimiento cicloidal de un disco de curvas (Figura 2.9) movido por una excéntrica solidaria al árbol de entrada. Por cada revolución de la excéntrica el disco de curvas avanza un saliente rodando sobre los rodillos exteriores. Este avance arrastra a su vez a los pernos del árbol de salida que describirán una cicloide dentro de los huecos del disco de curvas. La componente de traslación angular de este movimiento se corresponde con la rotación del árbol de salida. La relación de reducción viene, por tanto, determinada por el número de salientes. Para compensar los momentos de flexión y de las masas de cada disco en movimiento excéntrico, generalmente se utilizan dos discos desfasados entre sí 180°.

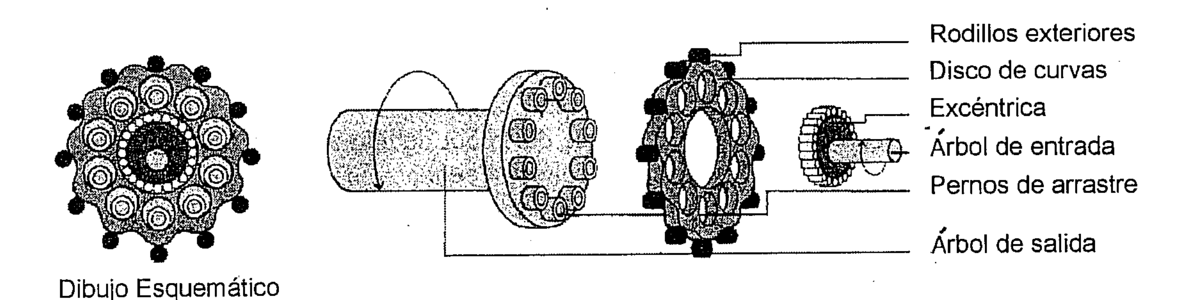


Figura 2.9. Reductor Cyclo. (Cortesía de Cyclo.)

Existe también un tercer tipo de reductor denominado REDEX-ACBAR, de funcionamiento similar al CYCLO, pero que presenta la posibilidad de poder reducir el juego angular desde el exterior a teóricamente un valor nulo.

2.2.3. Accionamiento directo

Como se ha indicado anteriormente, desde hace unos años existen en el mercado robots que poseen lo que se ha dado en llamar **accionamiento directo** (*Direct Drive* DD), en el que el eje del actuador se conecta directamente a la carga o articulación, sin la utilización de un reductor intermedio [AN-88] [YOUCEF-89] [MONTANO-88] . Este término suele utilizarse exclusivamente para robots con accionamiento eléctrico.

Este tipo de accionamiento aparece a raíz de la necesidad de utilizar robots en aplicaciones que exigen combinar gran precisión con alta velocidad. Los reductores introducen una serie de efectos negativos, como son juego angular, rozamiento o disminución de la rigidez del accionador, que pueden impedir alcanzar los valores de precisión y velocidad requeridos.

Las principales ventajas que se derivan de la utilización de accionamientos directos son las siguientes:

- Posicionamiento rápido y preciso, pues se evitan los rozamientos y juegos de las transmisiones y reductores.
- Aumento de las posibilidades de controlabilidad del sistema a costa de una mayor complejidad.
- Simplificación del sistema mecánico al eliminarse el reductor.

El principal problema que existe para la aplicación práctica de un accionamiento directo radica en el motor a emplear. Debe tratarse de motores que proporcionen un par elevado (unas 50-100 veces mayor que con reductor) a bajas revoluciones (las de movimiento de la articulación) manteniendo la máxima rigidez posible.

Entre los motores empleados para accionamiento directo y que cumplan estas características, se encuentran los motores síncronos y de continua sin escobillas (*brushless*), ambos con imanes permanentes fabricados con materiales especiales (samario-cobalto). También se utilizan motores de inducción de reluctancia variable. La necesaria utilización de este tipo de motores encarece notablemente el sistema de accionamiento.

Otra cuestión importante a tener en cuenta en el empleo de accionamientos directos es la propia cinemática del robot. Colocar motores, generalmente pesados y voluminosos, junto a las articulaciones, no es factible para todas las configuraciones del robot debido a las inercias que se generan. El estudio de la cinemática con la que se diseña el robot ha de tener en cuenta estos parámetros, estando la estructura final elegida altamente condicionada por ellos. Por este motivo, los robots de accionamiento directo son generalmente de tipo SCARA, cuyo diseño se corresponde bien con las necesidades que el accionamiento directo implica. Al eliminar el reductor también se disminuye de forma considerable la resolución real del codificador de posición acoplado al eje, tal y como se detalla en el epígrafe dedicado a sensores internos. Esto lleva a la utilización en los DD de codificadores de posición de muy alta resolución.

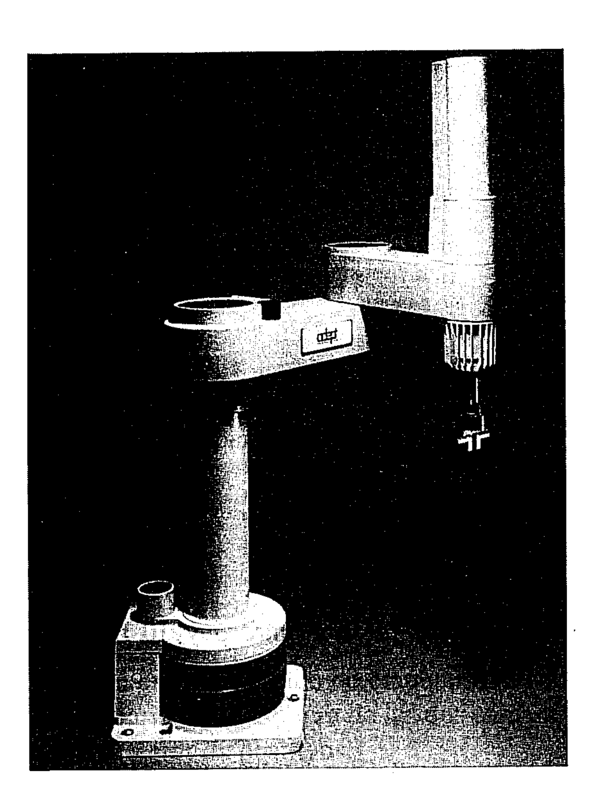


Figura 2.10. Robot AdeptThree de accionamiento directo. (Cortesía de Adept Technology.)

El primer robot comercial con accionamiento directo se presentó en 1984. Se trataba de un robot SCARA denominado AdeptOne, de la compañía norteamericana Adept Technology, Inc. A partir de entonces este tipo de robots se ha hecho popular para aplicaciones que requieran robots con altas prestaciones en velocidad y posicionamiento (montaje microelectrónico, corte de metal por láser, etc.). La Figura 2.10 muestra una fotografía del robot AdeptThree, de la misma compañía y también de accionamiento directo. En la actualidad un robot con accionamiento directo puede llegar a aumentar tanto la velocidad como la precisión de manera significativa con respecto a aquellos robots de accionamiento tradicional.

2.3. ACTUADORES

Los actuadores tienen por misión generar el movimiento de los elementos del robot según las órdenes dadas por la unidad de control. Los actuadores utilizados en robótica pueden emplear energía neumática, hidráulica o eléctrica. Cada uno de estos sistemas presenta características diferentes, siendo preciso

evaluarlas a la hora de seleccionar el tipo de actuador más conveniente. Las características a considerar son entre otras:

- Potencia.
- Controlabilidad.
- Peso y volumen.
- Precisión.
- Velocidad.
- Mantenimiento.
- Coste.

En este epígrafe se examinan los tres tipos de actuadores citados, comparándolos en cuanto a las características anteriores.

2.3.1. Actuadores neumáticos

En ellos la fuente de energía es aire a presión entre 5 y 10 bar. Existen dos tipos de actuadores neumáticos:

- Cilindros neumáticos.
- Motores neumáticos (de aletas rotativas o de pistones axiales).

En los primeros se consigue el desplazamiento de un émbolo encerrado en un cilindro, como consecuencia de la diferencia de presión a ambos lados de aquel (Figura 2.11). Los cilindros neumáticos pueden ser de simple o doble efecto. En los primeros, el émbolo se desplaza en un sentido como resultado del empuje ejercido por el aire a presión, mientras que en el otro sentido se desplaza como consecuencia del efecto de un muelle (que recupera al émbolo a su posición de reposo). En los cilin-

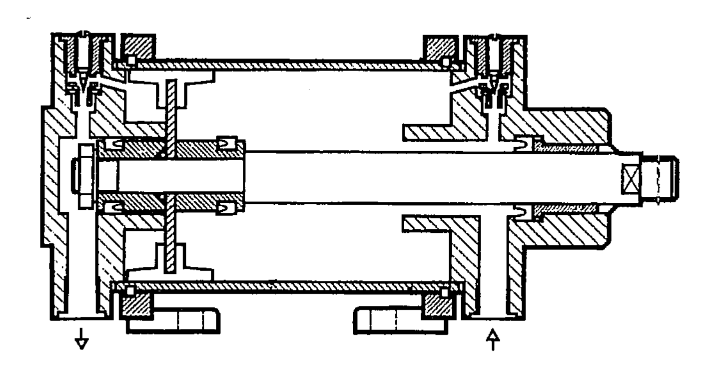
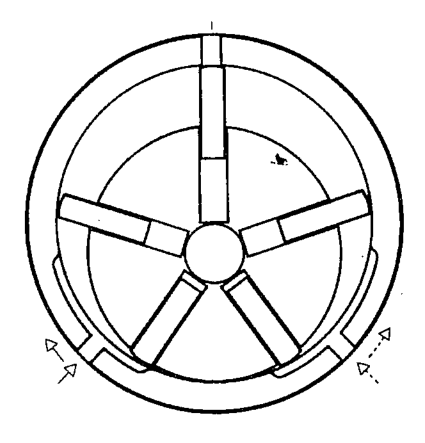


Figura 2.11. Esquema de cilindro neumático de doble efecto. (Cortesía de FESTO.)



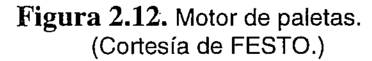


Figura 2.12. Motor de paletas.
(Cortesía de FESTO.)

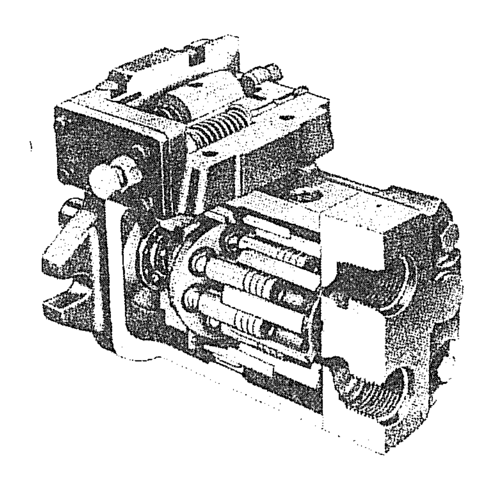


Figura 2.13. Motor de pistones axiales.
(Cortesía de CEDISA.)

dros de doble efecto el aire a presión es el encargado de empujar al émbolo en las dos direcciones, al poder ser introducido de forma arbitraria en cualquiera de las dos cámaras. En [DEPPERT-94] se explica con más detalle el funcionamiento de estos dispositivos.

Normalmente, con los cilindros neumáticos sólo se persigue un posicionamiento en los extremos del mismo y no un posicionamiento continuo. Esto último se puede conseguir con una válvula de distribución (generalmente de accionamiento eléctrico) que canaliza el aire a presión hacia una de las dos caras del émbolo alternativamente. Existen no obstante sistemas de posicionamiento continuo de accionamiento neumático, aunque debido a su coste y calidad todavía no resultan competitivos.

En los motores neumáticos se consigue el movimiento de rotación de un eje mediante aire a presión. Los dos tipos más usados son los motores de aletas rotativas y los motores de pistones axiales. En los primeros (Figura 2.12), sobre el rotor excéntrico están dispuestas las aletas de longitud variable. Al entrar aire a presión en uno de los compartimentos formados por dos aletas y la carcasa, éstas tienden a girar hacia una situación en la que el compartimento tenga mayor volumen. Los motores de pistones axiales tienen un eje de giro solidario a un tambor que se ve obligado a girar por las fuerzas que ejercen varios cilindros, que se apoyan sobre un plano inclinado (Figura 2.13).

Otro método común más sencillo de obtener movimientos de rotación a partir de actuadores neumáticos, se basa en el empleo de cilindros cuyo émbolo se encuentra acoplado a un sistema de piñón-cremallera. El conjunto forma una unidad compacta que puede adquirirse en el mercado como tal (Figura 2.14).

En general y debido a la compresibilidad del aire, los actuadores neumáticos no consiguen una buena precisión de posicionamiento. Sin embargo, su sencillez y robustez hacen adecuado su uso en aquellos casos en los que sea suficiente un posicionamiento en dos situaciones diferentes (todo o nada). Por ejemplo, son utilizados en manipuladores sencillos, en apertura y cierre de pinzas o en determinadas articulaciones de algún robot (como el movimiento vertical del tercer grado de libertad de algunos robots tipo SCARA).

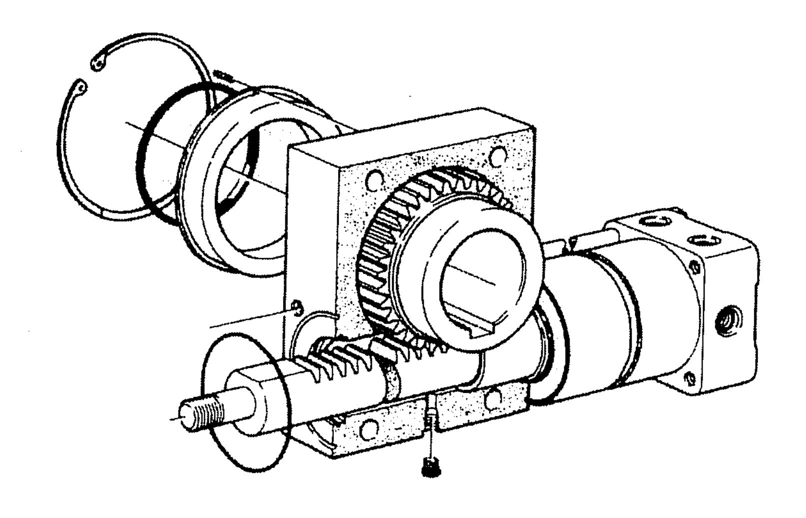


Figura 2.14. Actuador rotativo tipo piñón-cremallera. (Cortesía de REXROTH MECMAN.)

Siempre debe tenerse en cuenta que el empleo de un robot con algún tipo de accionamiento neumático deberá disponer de una instalación de aire comprimido, incluyendo: compresor, sistema de distribución (tuberías, electroválvulas), filtros, secadores, etc. No obstante, estas instalaciones neumáticas son frecuentes y existen en muchas de las fábricas donde se da cierto grado de automatización.

2.3.2. Actuadores hidráulicos

Este tipo de actuadores no se diferencian funcionalmente en mucho de los neumáticos. En ellos, en vez de aire se utilizan aceites minerales a una presión comprendida normalmente entre los 50 y 100 bar, llegándose en ocasiones a superar los 300 bar. Existen, como en el caso de los neumáticos, actuadores del tipo cilindro y del tipo motores de aletas y pistones [VOCKROTH-94].

Sin embargo, las características del fluido utilizado en los actuadores hidráulicos marcan ciertas diferencias con los neumáticos. En primer lugar, el grado de compresibilidad de los aceites usados es considerablemente inferior al del aire, por lo que la precisión obtenida en este caso es mayor. Por motivos similares, es más fácil en ellos realizar un control continuo, pudiendo posicionar su eje en todo un rango de valores (haciendo uso de servocontrol) con notable precisión. Además, las elevadas presiones de trabajo, diez veces superiores a las de los actuadores neumáticos, permiten desarrollar elevadas fuerzas y pares.

Por otra parte, este tipo de actuadores presenta estabilidad frente a cargas estáticas. Esto indica que el actuador es capaz de soportar cargas, como el peso o una presión ejercida sobre una superficie, sin aporte de energía (para mover el émbolo de un cilindro sería preciso vaciar éste de aceite). También es destacable su elevada capacidad de carga y relación potencia-peso, así como sus características de autolubricación y robustez.

Frente a estas ventajas existen también ciertos inconvenientes. Por ejemplo, las elevadas presiones a las que se trabaja propician la existencia de fugas de aceite a lo largo de la instalación. Asimismo, esta instalación es más complicada que la necesaria para los actuadores neumáticos y mucho más que para los eléctricos, necesitando de equipos de filtrado de partículas, eliminación de aire, sistemas de refrigeración y unidades de control de distribución.

Los accionamientos hidráulicos se usan con frecuencia en aquellos robots que deben manejar grandes cargas. Así, este tipo de accionamiento ha sido usado por robots como el UNIMATE 2000 y 4000 con capacidades de carga de 70 y 205 kg, respectivamente.

2.3.3. Actuadores eléctricos

Las características de control, sencillez y precisión de los accionamientos eléctricos ha hecho que sean los más usados en los robots industriales actuales.

Dentro de los actuadores eléctricos pueden distinguirse tres tipos diferentes [CORTÉS-89]:

- Motores de corriente continua (DC):
  - Controlados por inducido.
  - Controlados por excitación.
- Motores de corriente alterna (AC):
  - Síncronos.
  - Asíncronos.
- Motores paso a paso.

Motores de corriente continua (DC)

Son los más usados en la actualidad debido a su facilidad de control. En la Figura 2.15 se muestra un esquema de un motor DC seccionado, en el que se pueden apreciar sus distintos elementos. En este caso, el propio motor incluye un codificador de posición (encoder) para poder realizar su control.

Los motores DC están constituidos por dos devanados internos, inductor e inducido, que se alimentan con corriente continua:

- El inductor, también denominado devanado de excitación, está situado en el estator y crea un campo magnético de dirección fija, denominado de excitación.
- El inducido, situado en el rotor, hace girar al mismo debido a la fuerza de Lorentz que aparece como combinación de la corriente circulante por él y del campo magnético de excitación. Recibe la corriente del exterior a través del colector de delgas, en el que se apoyan unas escobillas de grafito.

Para que se pueda realizar la conversión de energía eléctrica en energía mecánica de forma continua es necesario que los campos magnéticos del estator y del rotor permanezcan estáticos entre sí. Esta transformación es máxima cuando ambos campos se encuentran en cuadratura. El colector de delgas es un conmutador sincronizado con el rotor encargado de que se mantenga el ángulo relativo entre el campo del estator y el creado por las corrientes rotóricas. De esta forma se consigue transfor-

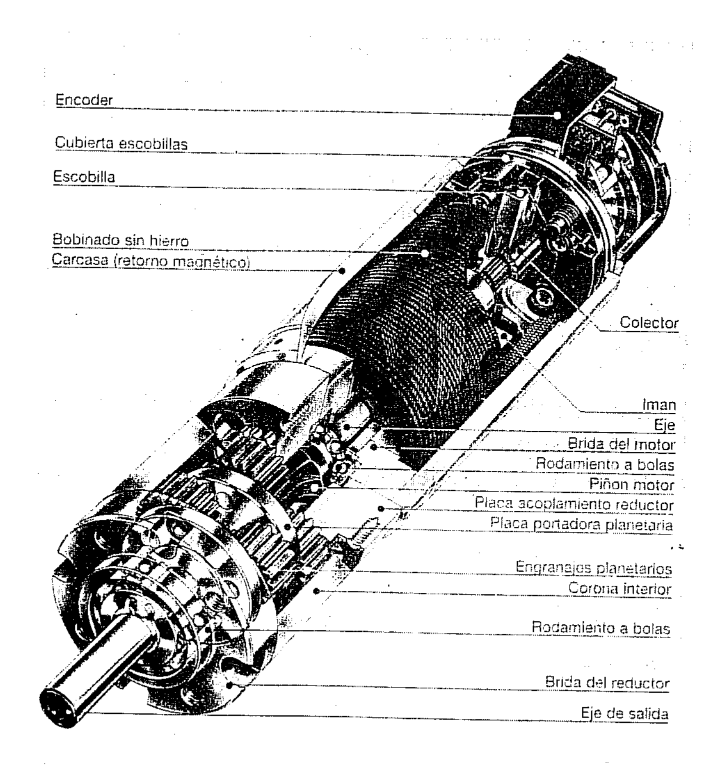


Figura 2.15. Esquema de motor DC seccionado con indicación de sus distintos componentes. (Cortesía de maxon motor.)

mar automáticamente, en función de la velocidad de la máquina, la corriente continua que alimenta al motor en corriente alterna de frecuencia variable en el inducido. Este tipo de funcionamiento se conoce con el nombre de *autopilotado*.

Al aumentar la tensión del inducido aumenta la velocidad de la máquina. Si el motor está alimentado a tensión constante, se puede aumentar la velocidad disminuyendo el flujo de excitación. Pero cuanto más débil sea el flujo, menor será el par motor que se puede desarrollar para una intensidad de inducido constante. En el caso de control por inducido, la intensidad del inductor se mantiene constante, mientras que la tensión del inducido se utiliza para controlar la velocidad de giro. En los controlados por excitación se actúa al contrario.

Del estudio de ambos tipos de motores, y realizándose las simplificaciones correspondientes, se obtiene que la relación entre tensión de control y velocidad de giro (función de transferencia), responde a un sistema de primer orden en los controlados por inducido, mientras que en el caso de los motores controlados por excitación, esta relación es la de un segundo orden (Figura 2.16). Ver el epígrafe 5.6.1 para más detalles.

Además, en los motores controlados por inducido se produce un efecto estabilizador de la velocidad de giro originado por la realimentación intrínseca que posee a través de la fuerza contraelectro-

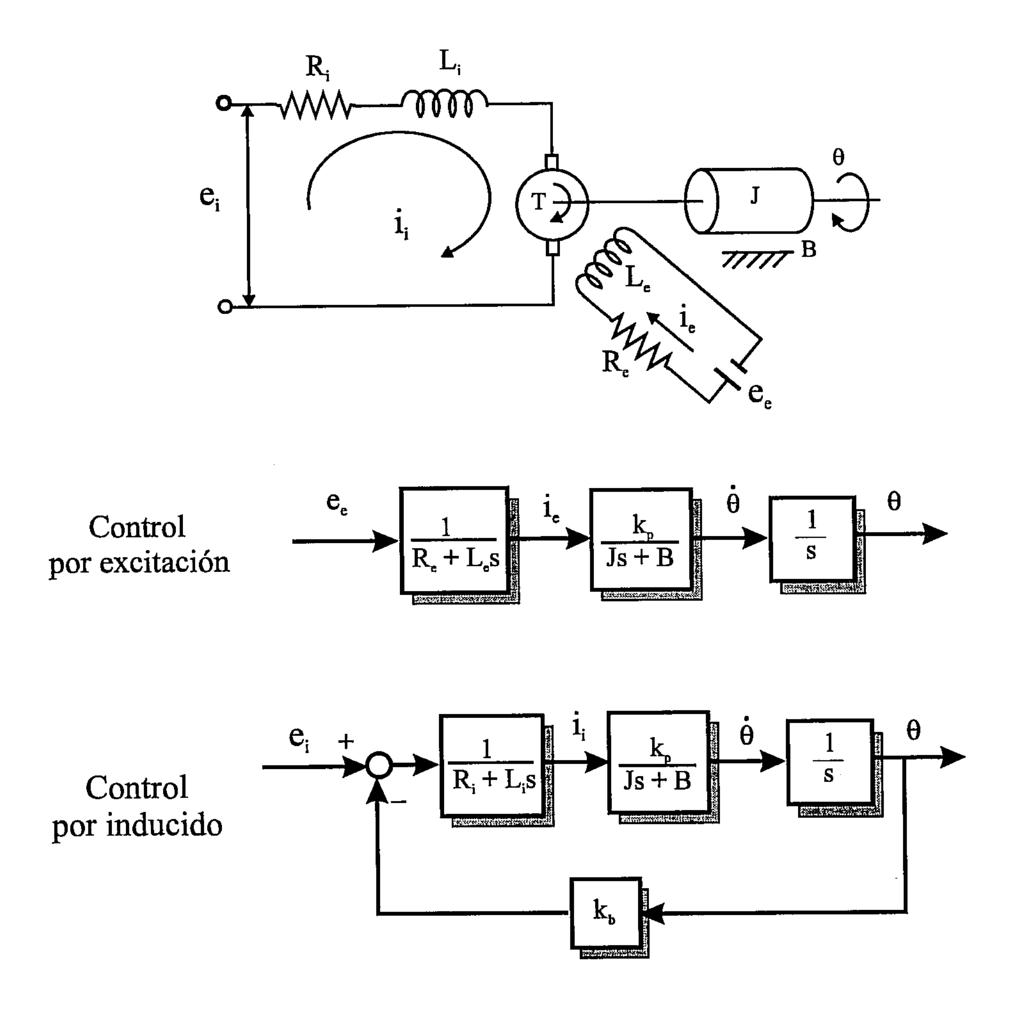


Figura 2.16. Motor DC. Esquema y funciones de transferencia.

motriz. Por estos motivos, de los dos tipos de motores DC es el controlado por inducido el que se usa en el accionamiento de robots.

Para mejorar el comportamiento de este tipo de motores, el campo de excitación se genera mediante imanes permanentes, con lo que se evitan fluctuaciones del mismo. Estos imanes son de aleaciones especiales como samario-cobalto. Además, para disminuir la inercia que poseería un rotor bobinado, que es el inducido, se construye éste mediante una serie de espiras serigrafiadas en un disco plano (ver Figura 2.17). En contrapartida, este tipo de rotor no posee apenas masa térmica lo que aumenta los problemas de calentamiento por sobrecarga.

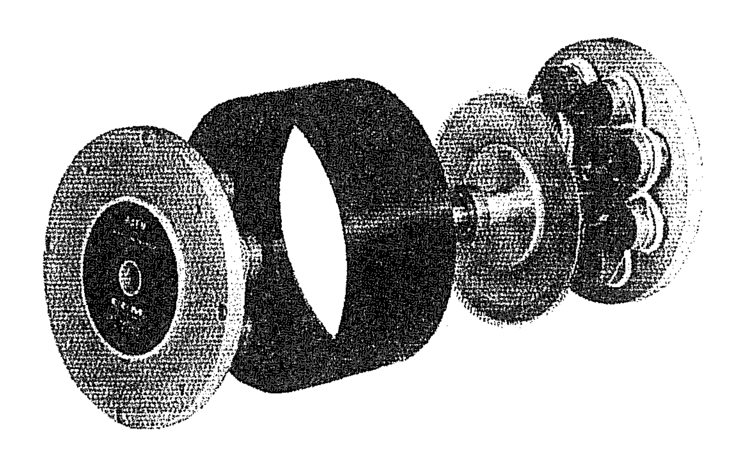


Figura 2.17. MotorDC de imanes permanentes controlado por inducido.

Las velocidades de rotación que se consiguen con estos motores son del orden de 1000 a 3000 r.p.m., con un comportamiento muy lineal y bajas constantes de tiempo. Las potencias que pueden manejar pueden llegar a los 10 kW.

Como se ha indicado, los motores DC son controlados mediante referencias de velocidad. Éstas normalmente son seguidas mediante un bucle de realimentación de velocidad analógico que se cierra mediante una electrónica específica (accionador del motor). Sobre este bucle de velocidad se coloca otro de posición, en el que las referencias son generadas por la unidad de control (microprocesador) en base al error entre la posición deseada y la real.

El motor de corriente continua presenta el inconveniente del obligado mantenimiento de las escobillas. Por otra parte, no es posible mantener el par con el rotor parado más de unos segundos, debido a los calentamientos que se producen en el colector.

Para evitar estos problemas, se han desarrollado en los últimos años motores sin escobillas (*brushless*). En éstos, los imanes de excitación se sitúan en el rotor y el devanado de inducido en el estator, con lo que es posible convertir la corriente mediante interruptores estáticos, que reciben la señal de conmutación a través de un detector de posición del rotor.

Motores paso a paso

Los motores paso a paso generalmente no han sido considerados dentro de los accionamientos industriales, debido principalmente a que los pares para los que estaban disponibles eran muy pequeños y los pasos entre posiciones consecutivas eran grandes. Esto limitaba su aplicación a controles de posición simples. En los últimos años se han mejorado notablemente sus características técnicas, especialmente en lo relativo a su control, lo que ha permitido fabricar motores paso a paso capaces de desarrollar pares suficientes en pequeños pasos para su uso como accionamientos industriales.

Existen tres tipos de motores paso a paso [BARRIENTOS-95]:

- De imanes permanentes.
- De reluctancia variable.
- Híbridos.

En los primeros, de imanes permanentes (Figura 2.18), el rotor, que posee una polarización magnética constante, gira para orientar sus polos de acuerdo al campo magnético creado por las fases del estator. En los motores de reluctancia variable, el rotor está formado por un material ferromagnético que tiende a orientarse de modo que facilite el camino de las líneas de fuerza del campo magnético generado por la bobinas de estator. Los motores híbridos combinan el modo de funcionamiento de los dos tipos anteriores.

En los motores paso a paso la señal de control son trenes de pulsos que van actuando rotativamente sobre una serie de electroimanes dispuestos en el estator. Por cada pulso recibido, el rotor del motor gira un determinado número discreto de grados.

Para conseguir el giro del rotor en un determinado número de grados, las bobinas del estator deben ser excitadas secuencialmente a una frecuencia que determina la velocidad de giro. Las inercias propias del arranque y parada (aumentadas por las fuerzas magnéticas en equilibrio que se dan cuando está parado) impiden que el rotor alcance la velocidad nominal instantáneamente, por lo que ésta, y por tanto la frecuencia de los pulsos que la fija, debe ser aumentada progresivamente.

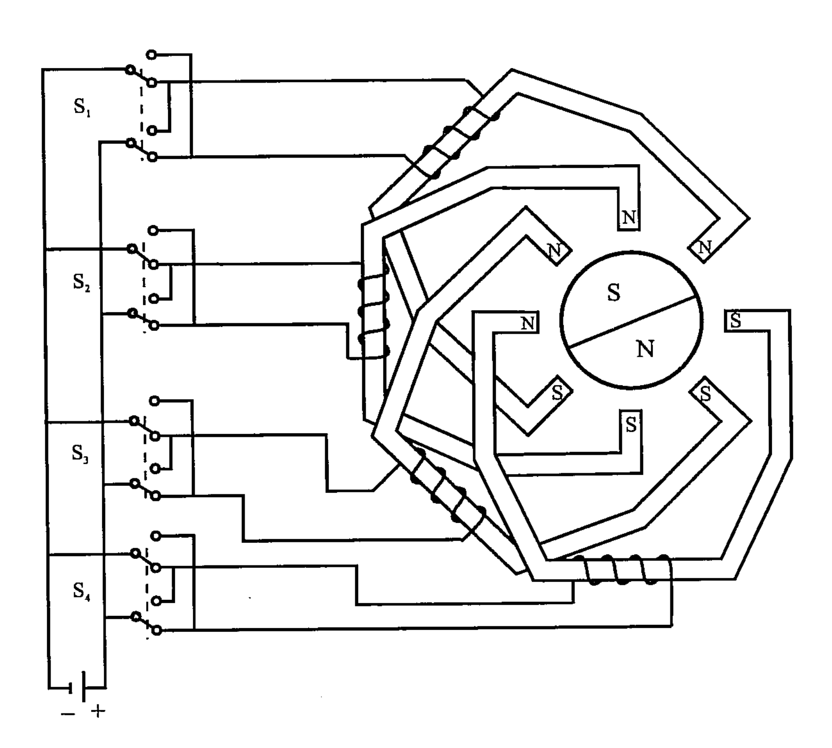


Figura 2.18. Esquema de un motor paso a paso de imanes permanentes con cuatro fases.

Para simplificar el control de estos motores existen circuitos especializados que a partir de tres señales (tren de pulsos, sentido de giro e inhibición) generan, a través de una etapa lógica, las secuencias de pulsos que un circuito de conmutación distribuye a cada fase.

Su principal ventaja con respecto a los servomotores tradicionales es su capacidad para asegurar un posicionamiento simple y exacto. Pueden girar además de forma continua, con velocidad variable, como motores síncronos, ser sincronizados entre sí, obedecer a secuencias complejas de funcionamiento, etc. Se trata al mismo tiempo de motores muy ligeros, fiables y fáciles de controlar, pues al ser cada estado de excitación del estator estable, el control se realiza en bucle abierto, sin la necesidad de sensores de realimentación.

Entre los inconvenientes se puede citar que su funcionamiento a bajas velocidades no es suave, y que existe el peligro de pérdida de una posición por trabajar en bucle abierto. Tienden a sobrecalentarse trabajando a velocidades elevadas y presentan un límite en el tamaño que pueden alcanzar.

Su potencia nominal es baja y su precisión (mínimo ángulo girado) llega típicamente hasta 1,8°. Se emplean para el posicionado de ejes que no precisan grandes potencias (giro de pinza) o para robots pequeños (educacionales); también son muy utilizados en dispositivos periféricos del robot, como mesas de coordenadas.

Motores de corriente alterna (AC)

Este tipo de motores no ha tenido aplicación en el campo de la robótica hasta hace unos años, debido fundamentalmente a la dificultad de su control. Sin embargo, las mejoras que se han introducido en las máquinas síncronas hacen que se presenten como un claro competidor de los motores de corriente continua. Esto se debe principalmente a tres factores:

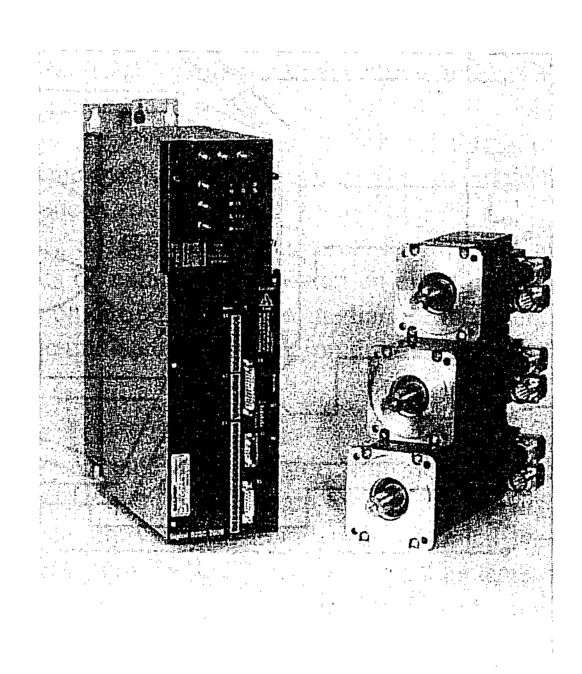


Figura 2.19. Gama de motores *brushless* con su etapa de control. (Cortesía de Baldor.)

- La construcción de rotores síncronos sin escobillas.
- Uso de convertidores estáticos que permiten variar la frecuencia (y así la velocidad de giro) con facilidad y precisión.
- Empleo de la microelectrónica que permite una gran capacidad de control.

El inductor se sitúa en el rotor y está constituido por imanes permanentes, mientras que el inducido, situado en el estator, está formado por tres devanados iguales decalados 120° eléctricos y se alimenta con un sistema trifásico de tensiones. Es preciso resaltar la similitud que existe entre este esquema de funcionamiento y el del motor sin escobillas (Figura 2.19).

En los motores síncronos la velocidad de giro depende únicamente de la frecuencia de la tensión que alimenta el inducido. Para poder variar ésta con precisión, el control de velocidad se realiza mediante un convertidor de frecuencia. Para evitar el riesgo de pérdida de sincronismo se utiliza un sensor de posición continuo que detecta la posición del rotor y permite mantener en todo momento el ángulo que forman los campos del estator y del rotor. Este método de control se conoce como autosíncrono o autopilotado.

El motor síncrono autopilotado excitado con imán permanente, también llamado motor senoidal, no presenta problemas de mantenimiento debido a que no posee escobillas y tiene una gran capacidad de evacuación de calor, ya que los devanados están en contacto directo con la carcasa. El control de posición se puede realizar sin la utilización de un sensor externo adicional, aprovechando el detector

Tabla 2.3. Características de distintos tipos de actuadores para robots

	Neumático	Hidráulico	Eléctrico
Energía	· Aire a presión (5-10 bar)	· Aceite mineral (50-100 bar)	· Corriente eléctrica
Opciones	· Cilindros · Motor de paletas · Motor de pistón	· Cilindros · Motor de paletas · Motor de pistones axiales	· Corriente continua · Corriente alterna · Motor paso a paso
Ventajas	· Baratos · Rápidos · Sencillos · Robustos	· Rápidos · Alta relación potencia-peso · Autolubricantes · Alta capacidad de carga · Estabilidad frente a cargas estáticas	· Precisos · Fiables · Fácil control · Sencilla instalación · Silenciosos
Desventajas	· Dificultad de control continuo · Instalación especial (compresor, filtros) · Ruidoso	· Difícil mantenimiento · Instalación especial (filtros, eliminación aire) · Frecuentes fugas · Caros	· Potencia limitada

de posición del rotor que posee el propio motor. Además permite desarrollar, a igualdad de peso, una potencia mayor que el motor de corriente continua. En la actualidad diversos robots industriales emplean este tipo de accionamientos con notables ventajas frente a los motores de corriente contínua.

En el caso de los motores asíncronos, no se ha conseguido resolver satisfactoriamente los problemas de control que presentan. Esto ha hecho que hasta el momento no tengan aplicación en robótica.

Como resumen de los tipos de actuadores empleados en robótica, en la Tabla 2.3 se presenta un cuadro comparativo de éstos.

2.4. SENSORES INTERNOS

Para conseguir que un robot realice su tarea con la adecuada precisión, velocidad e inteligencia, será preciso que tenga conocimiento tanto de su propio estado como del estado de su entorno. La información relacionada con su estado (fundamentalmente la posición de sus articulaciones) la consigue con los denominados sensores internos, mientras que la que se refiere al estado de su entorno, se adquiere con los sensores externos.

En este epígrafe se tratará únicamente de los sensores internos. Información sobre sensores externos puede verse en textos y bibliografía especializada en visión artificial [GONZALEZ-92], sensores de fuerza [CREUS-79], tacto [RUSSEL-90], distancia o telemetría. La información que la unidad de control del robot

Tabla 2.4. Tipos de sensores internos de robots

Presencia	· Inductivo · Capacitivo · Efecto hall · Célula Reed · Óptico · Ultrasonido · Contacto	
Posición	Analógicos	· Potenciómetros · Resolver · Sincro · Inductosyn · LVDT
	Digitales	· Encoders absolutos · Encoders incrementales · Regla óptica
Velocidad	· Tacogeneratriz	

puede obtener sobre el estado de su estructura mecánica es fundamentalmente la relativa a su posición y velocidad. En la Tabla 2.4 se resumen los sensores más comúnmente empleados para obtener información de presencia, posición y velocidad en robots industriales, explicados con más detalle en [PALLÁS-89] y [NACHTIGAL-90]. Se comentan a continuación los más significativos.

2.4.1. Sensores de posición

Para el control de posición angular se emplean fundamentalmente los denominados **encoders** y **resolvers**. Los potenciómetros dan bajas prestaciones por lo que no se emplean salvo en contadas ocasiones (robots educacionales, ejes de poca importancia).

Codificadores angulares de posición (encoders)

Los codificadores ópticos o encoders incrementales constan, en su forma más simple, de un disco transparente con una serie de marcas opacas colocadas radialmente y equidistantes entre sí; de un sistema de iluminación en el que la luz es colimada de forma correcta, y de un elemento fotorreceptor (Figura 2.20). El eje cuya posición se quiere medir va acoplado al disco trasparente. Con esta disposición, a medida que el eje gire se irán generando pulsos en el receptor cada vez que la luz atraviese cada marca, y llevando una cuenta de estos pulsos es posible conocer la posición del eje.

Existe, sin embargo, el problema de no saber si en un momento dado se está realizando un giro en un sentido o en otro, con el peligro que supone no estar contando adecuadamente. Una solución a este problema consiste en disponer de otra franja de marcas, desplazada de la anterior de manera que el tren de pulsos que con ella se genere esté desplazado 90° eléctricos con respecto al generado por la primera

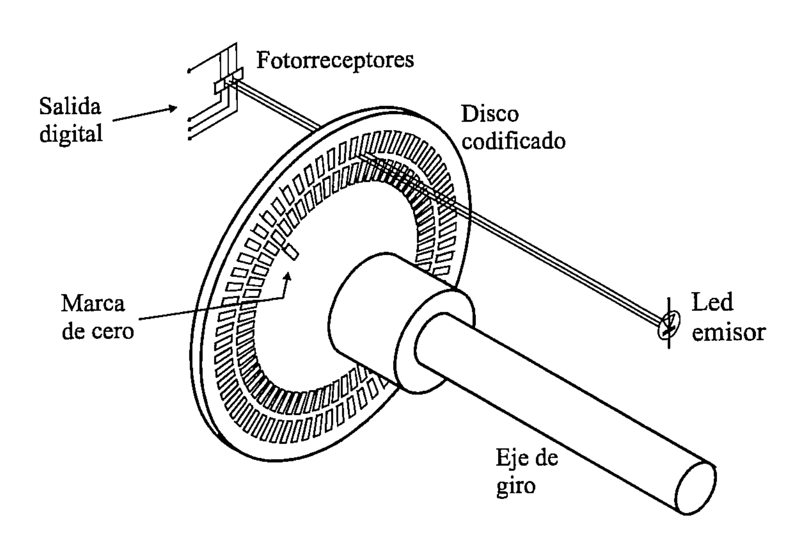


Figura 2.20. Disposición de un codificador óptico (encoder) incremental.

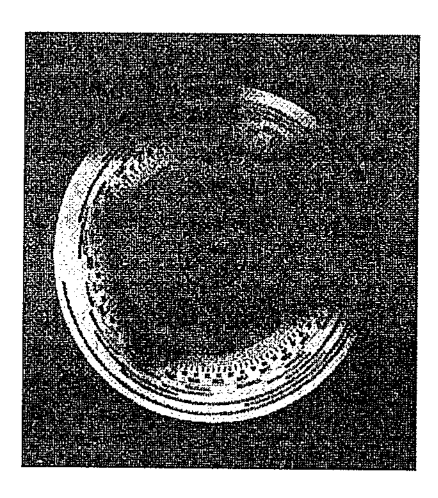


Figura 2.21. Disco codificado con código de Gray para utilizar en encoders absolutos. (Cortesía de HEIDENHAIN.)

franja. De esta manera, con un circuito relativamente sencillo es posible obtener una señal adicional que indique cuál es el sentido de giro, y que actúe sobre el contador correspondiente indicando que incremente o decremente la cuenta que se esté realizando.

Es necesario además disponer de una marca de referencia sobre el disco que indique que se ha dado una vuelta completa y que, por tanto, se ha de empezar la cuenta de nuevo. Esta marca sirve también para poder comenzar a contar tras recuperarse de una caída de tensión.

La resolución de este tipo de sensores depende directamente del número de marcas que se pueden poner físicamente en el disco. Un método relativamente sencillo para aumentar esta resolución es, no solamente contabilizar los flancos de subida de los trenes de pulsos, sino contabilizar también los de bajada, incrementando así por cuatro la resolución de captador, pudiéndose llegar, con ayuda de circuitos adicionales, hasta 100.000 pulsos por vuelta.

El funcionamiento básico de los codificadores o encoders absolutos es similar al de los incrementales. Se tiene una fuente de luz con las lentes de adaptación correspondientes, un disco graduado y unos fotorreceptores. En este caso, el disco transparente se divide en un número determinado de sectores (potencia de 2), codificándose cada uno de ellos según un código binario cíclico (normalmente código Gray) que queda representado por zonas transparentes y opacas dispuestas radialmente (Figura 2.21).

No es necesario ahora ningún contador o electrónica adicional para detectar el sentido de giro, pues cada posición (sector) es codificado de forma absoluta. Su resolución es fija, y vendrá dada por el número de anillos que posea el disco graduado. Resoluciones habituales van desde 2^8 a 2^{19} bits (desde 256 a 524.288 posiciones distintas).

Normalmente los sensores de posición se acoplan al eje del motor. Considerando que en la mayor parte de los casos entre el eje del motor y el de la articulación se sitúa un reductor de relación N, cada movimiento de la articulación se verá multiplicado por N al ser medido por el sensor. Éste aumentará así su resolución multiplicándola por N.

Este problema se soluciona en los encoders absolutos con la utilización de otro encoder absoluto más pequeño conectado por un engranaje reductor al principal, de manera que cuando éste gire un vuelta

completa, el codificado adicional avanzará una posición. Son los denominados encoders absolutos multivuelta.

Esta misma circunstancia originará que en el caso de los codificadores incrementales la señal de referencia o marca de cero, sea insuficiente para detectar el punto origen para la cuenta de pulsos, pues habrá N posibles puntos de referencia para un giro completo de la articulación. Para distinguir cuál de ellos es el correcto se suele utilizar un detector de presencia denominado de sincronismo, acoplado directamente al eslabón del robot que se considere. Cuando se conecta el robot desde una situación de apagado, es preciso ejecutar un procedimiento de búsqueda de referencias para los sensores (sincronizado). Durante su ejecución se leen los detectores de sincronismo que detectan la presencia o ausencia del eslabón del robot. Cuando se detecta la conmutación de presencia a ausencia de pieza, o viceversa, se atiende al encoder incremental, tomándose como posición de origen la correspondiente al primer pulso de marca de cero que aquel genere.

Los encoders pueden presentar problemas mecánicos debido a la gran precisión que se debe tener en su fabricación. La contaminación ambiental puede ser una fuente de interferencias en la transmisión óptica. Son dispositivos particularmente sensibles a golpes y vibraciones, estando su margen de temperatura de trabajo limitado por la presencia de componentes electrónicos.

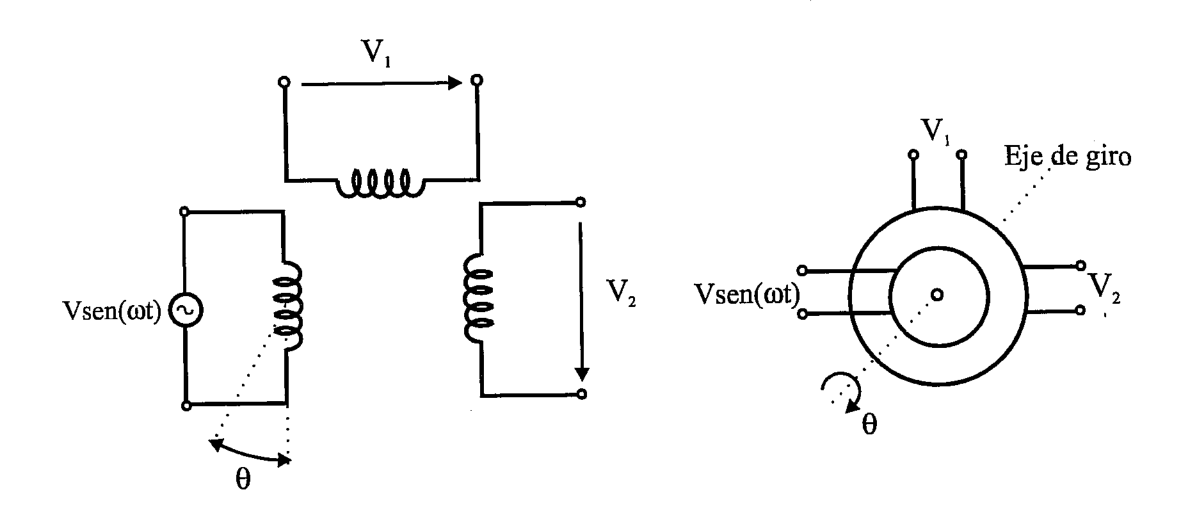


Figura 2.22. Esquema de funcionamiento de un resolver.

Captadores angulares de posición (sincro-resolvers)

La otra alternativa en sensores de posición para robots la representan los resolvers y los sincroresolvers, también llamados sincros. Se trata de captadores analógicos con resolución teóricamente infinita. El funcionamiento de los resolvers se basa en la utilización de una bobina solidaria al eje excitada por una portadora, generalmente con 400Hz, y por dos bobinas fijas situadas a su alrededor (Figura 2.22).

El giro de la bobina móvil hace que el acoplamiento con las bobinas fijas varíe, consiguiendo que la señal resultante en éstas dependa del seno del ángulo de giro. La bobina móvil excitada con tensión Vsen(ωt) y girada un ángulo θ induce en las bobinas fijas situadas en cuadratura las siguientes tensiones:

$$\begin{aligned} V_1 &= V \operatorname{sen}(\omega t) \operatorname{sen}\theta \\ V_2 &= V \operatorname{sen}(\omega t) \cos\theta \end{aligned} \qquad [2.\ 2]$$

que la llamada representación del ángulo θ en formato resolver.

El funcionamiento de los sincros es análogo al de los resolvers, excepto que las bobinas fijas forman un sistema trifásico en estrella. Para un giro θ de la bobina móvil excitada con tensión Vsen(ωt), admitiendo que los acoplamientos y los desfases son los mismos para todos los devanados, se obtienen las siguientes tensiones entre las fases del estator:

$$\begin{aligned} V_{13} &= \sqrt{3}\, V \cos(\omega t) \operatorname{sen}\theta \\ V_{32} &= \sqrt{3}\, V \cos(\omega t) \operatorname{sen}(\theta + 120^\circ) \\ V_{21} &= \sqrt{3}\, V \cos(\omega t) \operatorname{sen}(\theta + 240^\circ) \end{aligned} \qquad [2.\ 3]$$

que es la llamada representación del ángulo θ en formato sincro.

El cambio de formato sincro a formato resolver o viceversa es inmediato, ya que se puede pasar de uno a otro a través de la llamada red de Scott o transformador de Scott, de funcionamiento bidireccional.

Para poder tratar en el sistema de control la información generada por los resolvers y los sincros es necesario convertir las señales analógicas en digitales. Para ello se utilizan los llamados convertidores resolver/digital (R/D), que tradicionalmente se basan en dos tipos de estructuras distintas: seguimiento (tracking) y muestreo (sampling).

Ambos captadores son de tipo absoluto en cada vuelta del eje acoplado a ellos. Entre sus ventajas destacan su buena robustez mecánica durante el funcionamiento y su inmunidad a contaminación, humedad, altas temperaturas y vibraciones. Debido a su reducido momento de inercia, imponen poca carga mecánica al funcionamiento del eje.

La Tabla 2.5 presenta una comparación entre distintos tipos de sensores de posición angular atendiendo a diversos parámetros de funcionamiento. Se comenta brevemente a continuación alguno de ellos.

Tabla 2.5. Comparación entre distintos sensores de posición angular

	Robustez mecánica	Rango dinámico	Resolución	Estabilidad térmica
Encoder	mala	media	buena	buena
Resolver	buena	buena	buena	buena
Potenciometro	regular	mala	mala	mala

Dado el carácter continuo de la señal, la resolución de los resolvers es teóricamente infinita. Bien es verdad que depende en la mayoría de las ocasiones de una electrónica asociada, lo que limita la precisión de forma práctica. En el caso de los codificadores ópticos la resolución viene limitada por el número de secciones opaco/transparentes que se utilicen.

La exactitud estática, definida como la diferencia entre las posición física del eje y la señal eléctrica de salida, es relativamente alta tanto en resolvers como en codificadores ópticos. El rango dinámico se encuentra más limitado en el caso de los codificadores ópticos o digitales, no así en los resolvers donde con conversiones R/D adecuadas se puede trabajar con velocidades superiores a las 6000 rpm.

Sensores lineales de posición (LVDT e Inductosyn)

Entre los sensores de posición lineales destaca el transformador diferencial de variación lineal (LVDT) debido a su casi infinita resolución, poco rozamiento y alta repetibilidad. Su funcionamiento se basa en la utilización de un núcleo de material ferromagnético unido al eje cuyo movimiento se quiere medir. Este núcleo se mueve linealmente entre un devanado primario y dos secundarios, haciendo con su movimiento que varíe la inductancia entre ellos. La Figura 2.23 presenta un breve esquema de su funcionamiento.

Los dos devanados secundarios conectados en oposición serie ven cómo la inducción de la tensión alterna del primario, al variar la posición del núcleo, hace crecer la tensión en un devanado y disminuirla en el otro. Del estudio de la tensión E_0 se deduce que ésta es proporcional a la diferencia de inductancias mutuas entre el devanado primario con cada uno de los secundarios, y que por tanto depende linealmente del desplazamiento del vástago solidario al núcleo.

Además de las ventajas señaladas, el LVDT presenta una alta linealidad, gran sensibilidad y una respuesta dinámica elevada. Su uso está ampliamente extendido, a pesar del inconveniente de poder ser aplicado únicamente en la medición de pequeños desplazamientos.

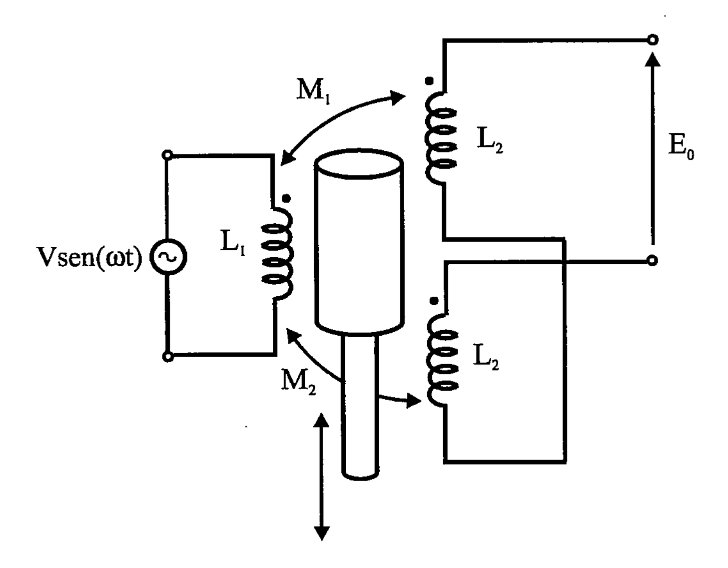


Figura 2.23. Esquema de funcionamiento de un LVDT.

Otros sensores lineales que también se emplean con relativa frecuencia son las denominadas reglas ópticas (equivalentes a los codificadores ópticos angulares) y las reglas magnéticas o Inductosyn (marca registrada de Farrand Industries Inc.). El funcionamiento del Inductosyn es similar al del resolver con la diferencia de que el rotor desliza linealmente sobre el estator, siendo la forma de los devanados la representada en la Figura 2.24. El estator se encuentra excitado por una tensión conocida que induce en el rotor dependiendo de su posición relativa una tensión V_s:

$$V_s = kV\cos\left(2\pi\frac{x}{P}\right) \qquad [2.\ 4]$$

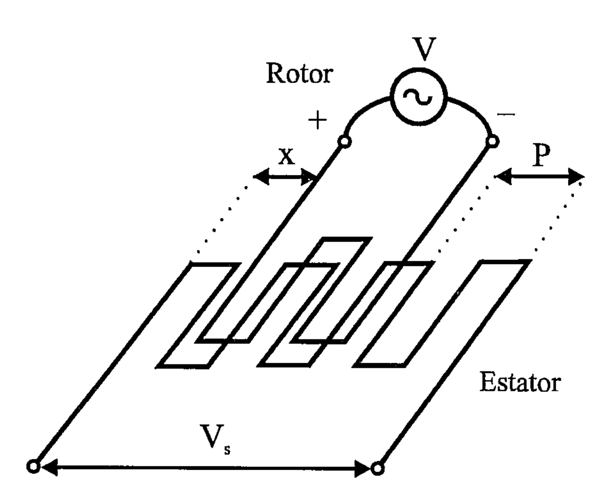


Figura 2.24. Esquema de funcionamiento de un Inductosyn.

2.4.2. Sensores de velocidad

La captación de la velocidad se hace necesaria para mejorar el comportamiento dinámico de los actuadores del robot. La información de la velocidad de movimiento de cada actuador (que tras el reductor es la de giro de la articulación) se realimenta normalmente a un bucle de control analógico implementado en el propio accionador del elemento motor. No obstante, en ocasiones en las que el sistema de control del robot lo exija, la velocidad de giro de cada actuador es llevada hasta la unidad de control del robot.

Normalmente, y puesto que el bucle de control de velocidad es analógico, el captador usado es una tacogeneratriz que proporciona una tensión proporcional a la velocidad de giro de su eje (valores típicos pueden ser 10 milivoltios por rpm).

Otra posibilidad, usada para el caso de que la unidad de control del robot precise valorar la velocidad de giro de las articulaciones, consiste en derivar la información de posición que ésta posee.

2.4.3. Sensores de presencia

Este tipo de sensor es capaz de detectar la presencia de un objeto dentro de un radio de acción determinado. Esta detección puede hacerse con o sin contacto con el objeto. En el segundo caso se utilizan diferentes principios físicos para detectar la presencia, dando lugar a los diferentes tipos de captadores (ver Tabla 2.4). En el caso de detección con contacto, se trata siempre de un interruptor, normalmente abierto o normalmente cerrado según interese, actuado mecánicamente a través de un vástago u otro dispositivo. Los detectores de presencia se utilizan en robótica principalmente como auxiliares de los detectores de posición, para indicar los límites de movimiento de las articulaciones y permitir localizar la posición de referencia de cero de éstos en el caso de que sean incrementales.

Además de esta aplicación, los sensores de presencia se usan como sensores externos, siendo muy sencillos de incorporar al robot por su carácter binario y su costo reducido. Los detectores inductivos permiten detectar la presencia o contar el número de objetos metálicos sin necesidad de contacto. Presentan el inconveniente de distinto comportamiento según del tipo de metal del que se trate. El mismo tipo de aplicación tiene los detectores capacitivos, más voluminosos, aunque en este caso los objetos a detectar no precisan ser metálicos. En cambio presentan problemas de trabajo en condiciones húmedas y con puestas a tierra defectuosas.

Los sensores basados en el efecto Hall detectan la presencia de objetos ferromagnéticos por la deformación que estos provocan sobre un campo magnético. Los captadores ópticos, sin embargo, pueden detectar la reflexión del rayo de luz procedente del emisor sobre el objeto.

Tabla 2.6. Sistemas de sujeción para robots

Tipos de sujeción	Accionamiento	Uso
Pinza de presión · desp. angular · desp. lineal	Neumático o eléctrico	Transporte y manipulación de piezas sobre las que no importe presionar.
Pinza de enganche	Neumático o eléctrico	Piezas de grandes dimensiones o sobre las que no se puede ejercer presión.
Ventosas de vacío	Neumático	Cuerpos con superficie lisa poco porosa (cristal, plástico, etc.).
Electroimán	Eléctrico	Piezas ferromagnéticas.

2.5. ELEMENTOS TERMINALES

Los elementos terminales, también llamados efectores finales (*end effector*) son los encargados de interaccionar directamente con el entorno del robot. Pueden ser tanto elementos de aprehensión como herramientas.

Si bien un mismo robot industrial es, dentro de unos límites lógicos, versátil y readaptable a una gran variedad de aplicaciones, no ocurre así con los elementos terminales, que son en muchos casos específicamente diseñados para cada tipo de trabajo.

Se puede establecer una clasificación de los elementos terminales atendiendo a si se trata de un elemento de sujeción o de una herramienta. Los primeros se pueden clasificar según el sistema de sujeción empleado. En la Tabla 2.6. se representan estas opciones, así como los usos más frecuentes.

Los elementos de sujeción se utilizan para agarrar y sostener los objetos y se suelen denominar pinzas. Se distingue entre las que utilizan dispositivos de agarre mecánico (Figura 2.25), y las que utilizan algún otro tipo de dispositivo (ventosas, pinzas magnéticas, adhesivas, ganchos, etc.).

En la elección o diseño de una pinza se han de tener en cuenta diversos factores. Entre los que afectan al tipo de objeto y de manipulación a realizar destacan el peso, la forma, el tamaño del objeto y la fuerza que es necesario ejercer y mantener para sujetarlo. Entre los parámetros de la pinza cabe destacar su peso (que afecta a las inercias del robot), el equipo de accionamiento y la capacidad de control.

El accionamiento neumático es el más utilizado por ofrecer mayores ventajas en simplicidad, precio y fiabilidad, aunque presenta dificultades de control de posiciones intermedias. En ocasiones se utilizan accionamientos de tipo eléctrico.

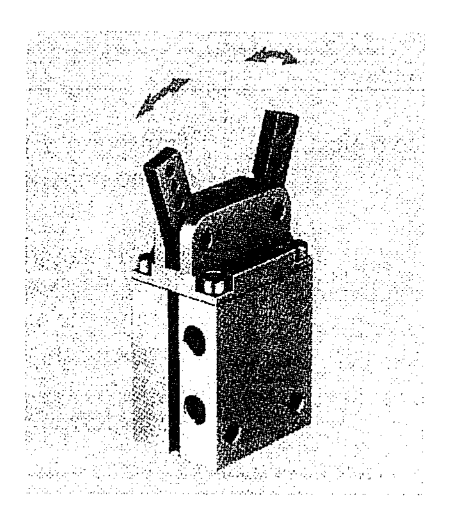


Figura 2.25. Pinza neumática de dedos paralelos. (Cortesía de FESTO.)

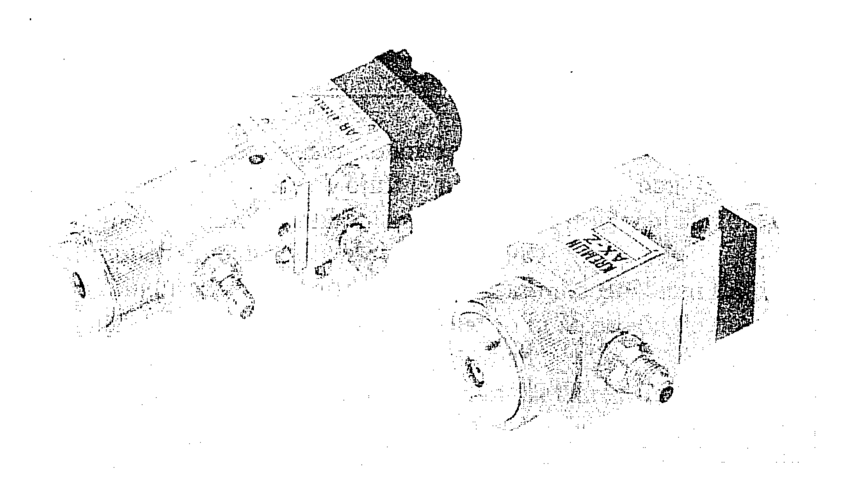


Figura 2.26. Pistolas neumáticas de pulverización de pintura. (Cortesía de Kremlin.)

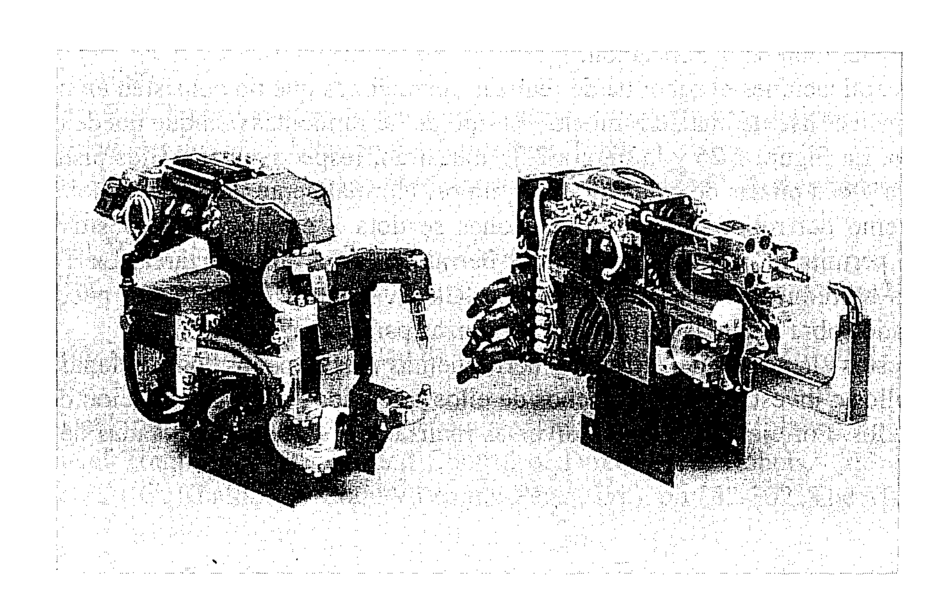


Figura 2.27. Pinzas de soldadura con transformador incorporado.
De accionamiento por tijera y rectilíneo. (Cortesía de ARO.)

En la pinza se suelen situar sensores para detectar el estado de la misma (abierto o cerrado). Se pueden incorporar a la pinza otro tipo de sensores para controlar el estado de la pieza, sistemas de visión que proporcionen datos geométricos de los objetos, detectores de proximidad, sensores fuerza-par, etc.

Tabla 2.7. Herramientas terminales para robots

Tipo de herramienta	Comentarios
Pinza soldadura por puntos	Dos electrodos que se cierran sobre la pieza a soldar.
Soplete soldadura al arco	Aportan el flujo de electrodo que se funde.
Cucharón para colada	Para trabajos de fundición.
Atornillador	Suelen incluir la alimentación de tornillos.
Fresa-lija	Para perfilar, eliminar rebabas, pulir, etc.
Pistola de pintura	Por pulverización de la pintura.
Cañón láser	Para corte de materiales, soldadura o inspección.
Cañón de agua a presión	Para corte de materiales.

Como se ha indicado, el elemento terminal de aprehensión debe ser diseñado con frecuencia a medida para la aplicación. Existen ciertos elementos comerciales que sirven de base para la pinza, siendo posible a partir de ellos diseñar efectores válidos para cada aplicación concreta. Sin embargo, en otras ocasiones el efector deber ser desarrollado íntegramente, constituyendo su coste un porcentaje importante dentro del total de la aplicación.

En muchas aplicaciones el robot ha de realizar operaciones que no consisten en manipular objetos, sino que implican el uso de una herramienta. El tipo de herramientas con que puede dotarse a un robot es muy amplio. La Figura 2.26 y la Figura 2.27 muestran, respectivamente, dos pistolas de pulverización de pintura y dos pinzas de soldadura por puntos. Normalmente, la herramienta está fijada rígidamente al extremo del robot aunque en ocasiones se dota a éste de un dispositivo de cambio automático, que permita al robot usar diferentes herramientas durante su tarea. La Tabla 2.7 enumera algunas de las herramientas más frecuentes. En [GROOVER-89] y [FERRATÉ-86] puede encontrarse un estudio detallado sobre distintas herramientas y pinzas para robots.

Aparte de estos elementos de sujeción y herramientas más o menos convencionales, existen interesantes desarrollos e investigaciones, muchos de ellos orientados a la manipulación de objetos complicados y delicados. Por ejemplo, existen diversas realizaciones de pinzas dotadas de tacto o de dedos con falanges.

2.6. BIBLIOGRAFÍA

[AN-88] An, C.H., Atkenson, C.G. y Hollerbach, J.M., "Direct Drive Arms", en: *Model-based Control of a Robot Manipulator*, The MIT Press, 1988.

[BARRIENTOS-95] Barrientos, A., Peñín, L.F., Carrera, J., *Automatización de la fabricación. Transductores, actuadores y autómatas programables*, Sección de Publicaciones ETSI Industriales de Madrid, Universidad Politécnica de Madrid, 1995.

[CORTÉS-89] Cortés Cherta, M., *Curso moderno de máquinas eléctricas rotativas*. Cuatro tomos. Editores Técnicos Asociados, Barcelona, 1989.

[CODINA-92] Codina, E., "Servoactuadores neumáticos", *Automática e Instrumentación,* 224, pp 118-129, 1992.

[CODINA-90] Codina, E., "Servoactuadores oleohidráulicos", *Automática e Instrumentación*, 203, pp 171-182, 1990.

[CREUS-79] Creus, A., *Instrumentación industrial*, Marcombo Boixareu Editores, Barcelona, 1979.

[CYCLO-96] Cyclo Getriebebau, *Catálogo de productos*, 1992.

[DEPPERT-94] Deppert, W., Dispositivos neumáticos, Marcombo Boixareu Editores, Barcelona, 1994.

[FERRANDO-93] Ferrando, F., J. Mercader, "Criterios para la selección de un reductor de velocidad", *Automática e Instrumentación*, 237, pp 100-106, 1993.

[FERRATÉ-86] Ferraté, G., *Robótica industrial*, Marcombo Boixareu Editores, Barcelona, 1986.

[GONZALEZ-92] R.C Gonzalez y R.E. Woods, *Digital Image Processing*, Addison Wesley, 1992.

[GROOVER-89] Groover, M.P, "Robot y efectores finales", en: *Robótica Industrial. Tecnología, Programación y Aplicaciones*, McGraw-Hill, Madrid, 1989.

[HARMONIC-96] HarmonicDrive, *Catálogo de Productos*, 1990.

[MONTANO-88] Montano, L., Tardós, J.D., y Sagüés, C., "Accionamientos directos para robots", *Automática e Instrumentación*, 179, pp 251-261, 1988.

[NACHTIGAL-90] Natchtigal, C.L., *Instrumentation and control. Fundamentals and applications*, Wiley-Interscience, 1990.

[PALLÁS-89] Pallás, R., "Transductores de reactancia variable y electromagnéticos", en: *Transductores y acondicionadores de señal*, Marcombo, Barcelona, 1989.

[PRENYANOSA-94] Prenyanosa, G., "Sensores de posición y velocidad", *Automática e Instrumentación,* 244, pp 66-82,1994.

[RUSSEL-90] Russel, R.A., *Robot Tactile Sensing*, Prentice-Hall, 1990.

[SUDRIÁ-95] Sudriá, A., "Servomotores eléctricos", *Automática e Instrumentación*, 259, pp. 86-97, 1995

[SUDRIÁ-88] Sudriá, A., "Control de velocidad de motores eléctricos", *Automática e Instrumentación*, 182, pp. 105-136, 1988.

[VOCKROTH-94] Vockroth, R.W., *Industrial Hydraulics*, Delmar Publishers, Nueva York, 1994.

[YOUCEF-89] Youcef-Toumi, K., "Design and Control of Direct-Drive Robots - a Survey", en: *The Robotics Review 1* (O. Khatib, J.J. Craig y Lozano-Pérez, Ed.), pp 283-302. The MIT Press, Cambridge, 1989.



Capítulo 3

HERRAMIENTAS MATEMÁTICAS PARA LA LOCALIZACIÓN ESPACIAL

De pequeña centella se levanta gran fuego.
Refranero popular

La manipulación de piezas llevada a cabo por un robot implica el movimiento espacial de su extremo. Asimismo, para que el robot pueda recoger una pieza, es necesario conocer la posición y orientación de ésta con respecto a la base del robot. Se aprecia entonces la necesidad de contar con una serie de herramientas matemáticas que permitan especificar la posición y orientación en el espacio de piezas, herramientas y, en general, de cualquier objeto.

Estas herramientas han de ser lo suficientemente potentes como para permitir obtener de forma sencilla relaciones espaciales entre distintos objetos y en especial entre éstos y el manipulador. Los siguientes epígrafes introducen de forma progresiva estas herramientas, de especial importancia para la adecuada comprensión de desarrollos en capítulos posteriores. Sin embargo, es necesario resaltar que éstas son de aplicación general para el tratamiento de problemas de localización espacial y que, por tanto, no son de aplicación exclusiva en el campo de la robótica.

Los dos primeros apartados presentan los distintos métodos existentes para la representación de la posición y orientación espacial de un cuerpo rígido. Los conceptos se irán introduciendo por orden creciente de dificultad, comenzando con la representación en dos dimensiones, para seguidamente pasar al análisis en tres. En el siguiente epígrafe se introduce el concepto de matriz de transformación homogénea, necesaria para la representación conjunta de posición y orientación, junto con sus propiedades y aplicaciones. Se trata de una herramienta muy útil para representar transformaciones espaciales, estando su uso ampliamente extendido en diversos campos además del de la robótica, como por ejemplo en el de gráficos por computador.

Los denominados cuaternios, al tratarse de una herramienta de uso más restringido, no son analizados con el suficiente detalle en la bibliografía existente. Se trata de un método de gran economía computacional utilizado incluso por algunos robots comerciales para la representación de orientación, y por ello se ha incluido un apartado dedicado a su estudio.

3.1. REPRESENTACIÓN DE LA POSICIÓN

Para localizar un cuerpo rígido en el espacio es necesario contar con una herramienta que permita la localización espacial de sus puntos. En un plano el posicionamiento tiene dos grados de libertad, y por tanto la posición de un punto vendrá definida por dos componentes independientes. En el caso de un espacio tridimensional será necesario emplear tres componentes.

La forma más intuitiva y utilizada de especificar la posición de un punto son coordenadas cartesianas. Existen además otros métodos, igualmente válidos, y también ampliamente extendidos, como son las coordenadas polares para dos dimensiones, y las cilíndricas y esféricas para espacios de tres dimensiones.

3.1.1. Sistema cartesiano de referencia

Normalmente los sistemas de referencia se definen mediante ejes perpendiculares entre sí con un origen definido. Estos se denominan **sistemas cartesianos**, y en el caso de trabajar en el plano (2 dimensiones),

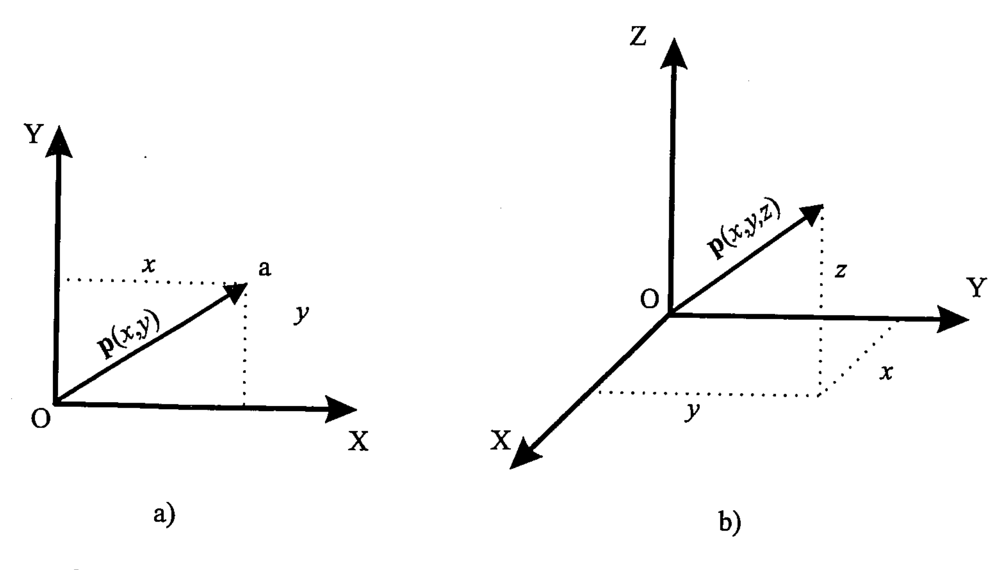


Figura 3.1. Representación de un vector en coordenadas cartesianas en 2 y 3 dimensiones.

el sistema de referencia OXY correspondiente queda definido por dos vectores coordenados OX y OY perpendiculares entre sí con un punto de intersección común O. (Figura 3.1-a).

Si se trabaja en el espacio (tres dimensiones), el sistema cartesiano OXYZ está compuesto por una terna ortonormal de vectores coordenados OX, OY y OZ, tal y como se ve en la Figura 3.1-b. Se trata de una terna ortonormal a derechas.

3.1.2. Coordenadas cartesianas

Si se trabaja en un plano, con su sistema coordenado OXY de referencia asociado, un punto a vendrá expresado por las componentes (x,y) correspondientes a los ejes coordenados del sistema OXY. Este punto tiene asociado un vector $\mathbf{p}(x,y)$, que va desde el origen O del sistema OXY hasta el punto a (ver Figura 3.1-a). Por tanto, la posición del extremo del vector **p** está caracterizado por las dos componentes (x,y), denominadas **coordenadas cartesianas** del vector y que son las proyecciones del vector **p** sobre los ejes OX y OY.

En el caso de que se trabaje en tres dimensiones, un vector viene definido con respecto al sistema de referencia OXYZ mediante las coordenadas correspondientes a cada uno de los ejes coordenados. En el caso de la Figura 3.1-b, el vector **p** estará definido por las componentes cartesianas (x,y,z).

3.1.3. Coordenadas polares y cilíndricas

Para un plano, es posible también caracterizar la localización de un punto o vector **p** respecto a un sistema de ejes cartesianos de referencia OXY utilizando las denominadas **coordenadas polares** $\mathbf{p}(r,\theta)$ (Figura 3.2-a). En esta representación, r representa la distancia desde el origen O del sistema hasta el

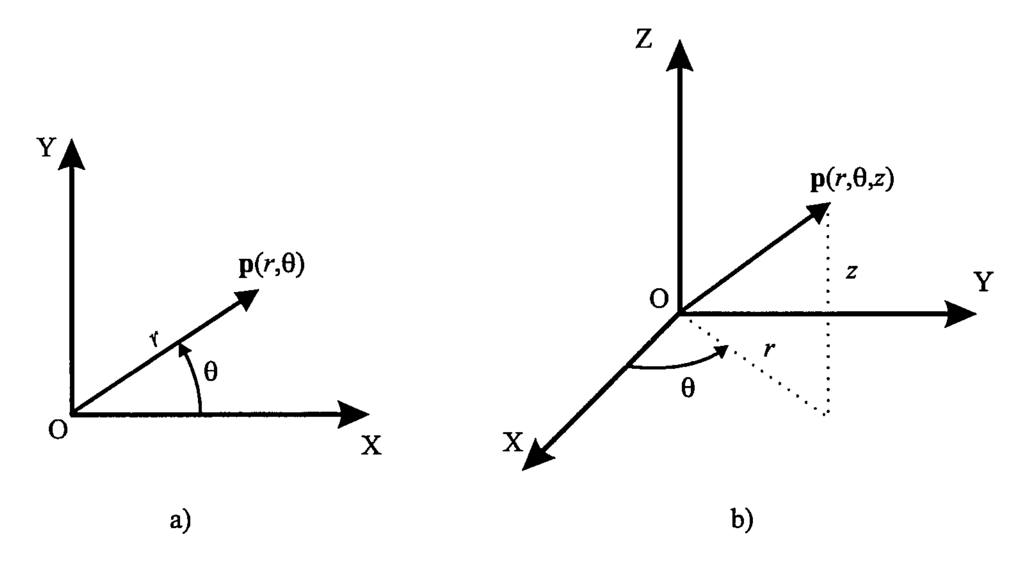


Figura 3.2. Representación de a) coordenadas polares y b) cilíndricas.

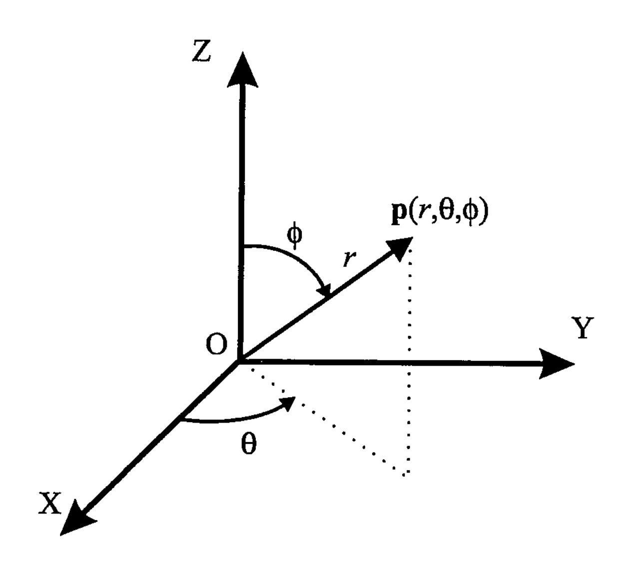


Figura 3.3. Representación de un vector en coordenadas esféricas.

extremo del vector **p**, mientras que θ es el ángulo que forma el vector **p** con el eje OX.

En el caso de trabajar en tres dimensiones, un vector **p** podrá expresarse con respecto a un sistema de referencia OXYZ, mediante las **coordenadas cilíndricas p**(r,θ,z) (Figura 3.2-b). Las componentes r y θ tienen el mismo significado que en el caso de coordenadas polares, aplicado el razonamiento sobre el plano OXY, mientras que la componente z expresa la proyección sobre el eje OZ del vector **p**.

3.1.4. Coordenadas esféricas

También es posible utilizar coordenadas esféricas para realizar la localización de un vector en un espacio de tres dimensiones. Utilizando el sistema de referencia OXYZ, el vector **p** tendrá como **coordenadas esféricas** (r,θ,ϕ), donde la componente r es la distancia desde el origen O hasta el extremo del vector **p**; la componente θ es el ángulo formado por la proyección del vector **p** sobre el plano OXY con el eje OX; y la componente φ es el ángulo formado por el vector **p** con el eje OZ. (Figura 3.3.)

3.2. REPRESENTACIÓN DE LA ORIENTACIÓN

Un punto queda totalmente definido en el espacio a través de los datos de su posición. Sin embargo, para el caso de un sólido, es necesario además definir cuál es su orientación con respecto a un sistema de referencia. En el caso de un robot, no es suficiente con especificar cuál debe ser la posición de su

extremo, sino que en general, es también necesario indicar su orientación. Por ejemplo, en el caso de un robot que tenga que realizar sobre una pieza curva una operación de pulido, no bastaría con especificar los puntos de la superficie para situar adecuadamente la herramienta, sino que será necesario también conocer la orientación con que la herramienta ha de realizar la operación.

Una orientación en el espacio tridimensional viene definida por tres grados de libertad o tres componentes linealmente independientes. Para poder describir de forma sencilla la orientación de un objeto respecto a un sistema de referencia, es habitual asignar solidariamente al objeto un nuevo sistema, y después estudiar la relación espacial existente entre los dos sistemas. De forma general, esta relación vendrá dada por la posición y orientación del sistema asociado al objeto respecto al de referencia. Para el análisis de los distintos métodos de representar orientaciones se supondrá que ambos sistemas coinciden en el origen, y que por tanto no existe cambio alguno de posición entre ellos.

3.2.1. Matrices de rotación

Las matrices de rotación son el método más extendido para la descripción de orientaciones, debido principalmente a la comodidad que proporciona el uso del álgebra matricial.

Supóngase que se tiene en el plano dos sistemas de referencia OXY y OUV con un mismo origen O, siendo el sistema OXY el de referencia fijo y el sistema OUV el móvil solidario al objeto (Figura 3.4-a). Los vectores unitarios de los ejes coordenados del sistema OXY son $\mathbf{i}_x$, $\mathbf{j}_y$, mientras que los del sistema OUV son $\mathbf{i}_u$, $\mathbf{j}_v$.

Un vector **p** del plano se puede representar en ambos sistemas como:

$$\begin{aligned} \mathbf{p}_{xy} &= \left[p_x, p_y\right]^T = p_x \cdot \mathbf{i}_x + p_y \cdot \mathbf{j}_y \\ \mathbf{p}_{uv} &= \left[p_u, p_v\right]^T = p_u \cdot \mathbf{i}_u + p_v \cdot \mathbf{j}_v \end{aligned} \qquad [3.\ 1]$$

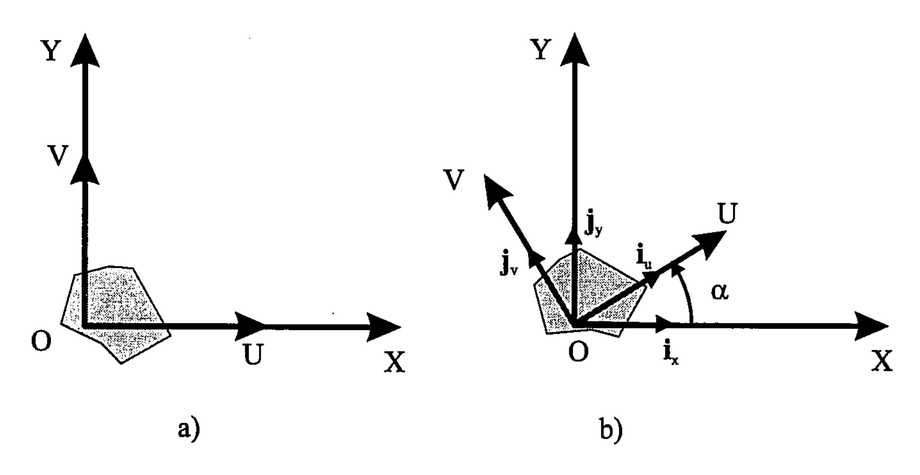


Figura 3.4. Orientación de un sistema OUV respecto a otro OXY en un plano.

Realizando una sencilla serie de transformaciones se puede llegar a la siguiente equivalencia:

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \mathbf{R} \begin{bmatrix} p_u \\ p_v \end{bmatrix} \qquad [3.\ 2]$$

donde:

$$\mathbf{R} = \begin{bmatrix} \mathbf{i}_x \mathbf{i}_u & \mathbf{i}_x \mathbf{j}_v \\ \mathbf{j}_y \mathbf{i}_u & \mathbf{j}_y \mathbf{j}_v \end{bmatrix} \qquad [3.\ 3]$$

es la llamada **matriz de rotación**, que define la orientación del sistema OUV con respecto al sistema OXY, y que sirve para transformar las coordenadas de un vector en un sistema a las del otro. También recibe el nombre de matriz de cosenos directores. Es fácil de comprobar que se trata de una matriz ortonormal, tal que $\mathbf{R}^{-1} = \mathbf{R}^T$.

En el caso de dos dimensiones, la orientación viene definida por un único parámetro independiente. Si se considera la posición relativa del sistema OUV girado un ángulo α sobre el OXY (Figura 3.4-b), tras realizar los correspondientes productos escalares, la matriz **R** será de la forma:

$$\mathbf{R} = \begin{bmatrix} \cos\alpha & -\operatorname{sen}\alpha \\ \operatorname{sen}\alpha & \cos\alpha \end{bmatrix} \qquad [3.\ 4]$$

Para el caso en que $\alpha = 0$, en el que los ejes coordenados de ambos sistemas coinciden, la matriz **R** corresponderá a la matriz unitaria.

En un espacio tridimensional, el razonamiento a seguir es similar. Supónganse los sistemas OXYZ y OUVW, coincidentes en el origen, siendo el OXYZ el sistema de referencia fijo, y el OUVW el solidario al objeto cuya orientación se desea definir. (Figura 3.5-a). Los vectores unitarios del sistema OXYZ serán $\mathbf{i}_x$, $\mathbf{j}_y$, $\mathbf{k}_z$, mientras que los del OUVW serán $\mathbf{i}_u$, $\mathbf{j}_v$, $\mathbf{k}_w$. Un vector **p** del espacio podrá ser referido a cualquiera de los sistemas de la siguiente manera:

$$\begin{aligned} \mathbf{p}_{uvw} &= [p_u, p_v, p_w]^T = p_u \cdot \mathbf{i}_u + p_v \cdot \mathbf{j}_v + p_w \cdot \mathbf{k}_w \\ \mathbf{p}_{xyw} &= [p_x, p_y, p_z]^T = p_x \cdot \mathbf{i}_x + p_y \cdot \mathbf{j}_y + p_z \cdot \mathbf{k}_z \end{aligned} \qquad [3.\ 5]$$

Y al igual que en dos dimensiones, se puede obtener la siguiente equivalencia:

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \mathbf{R} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \qquad [3.\ 6]$$

donde:

$$\mathbf{R} = \begin{bmatrix} \mathbf{i}_x\mathbf{i}_u & \mathbf{i}_x\mathbf{j}_v & \mathbf{i}_x\mathbf{k}_w \\ \mathbf{j}_y\mathbf{i}_u & \mathbf{j}_y\mathbf{j}_v & \mathbf{j}_y\mathbf{k}_w \\ \mathbf{k}_z\mathbf{i}_u & \mathbf{k}_z\mathbf{j}_v & \mathbf{k}_z\mathbf{k}_w \end{bmatrix} \qquad [3.7]$$

es la **matriz de rotación** que define la orientación del sistema OUVW con respecto al sistema OXYZ. Al igual que en dos dimensiones, también recibe el nombre de matriz de cosenos directores y se trata de un matriz ortonormal, tal que la inversa de la matriz **R** es igual a su traspuesta: $\mathbf{R}^{-1} = \mathbf{R}^T$.

La principal utilidad de esta matriz de rotación corresponde a la representación de la orientación de sistemas girados únicamente sobre uno de los ejes principales del sistema de referencia.

En la Figura 3.5-b, la orientación del sistema OUVW, con el eje OU coincidente con el eje OX, vendrá representada mediante la matriz:

$$\mathbf{R}(\mathbf{x},\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\mathrm{sen}\alpha \\ 0 & \mathrm{sen}\alpha & \cos\alpha \end{bmatrix} \qquad [3.8]$$

En la Figura 3.6-a, la orientación del sistema OUVW, con el eje OV coincidente con el eje OY,

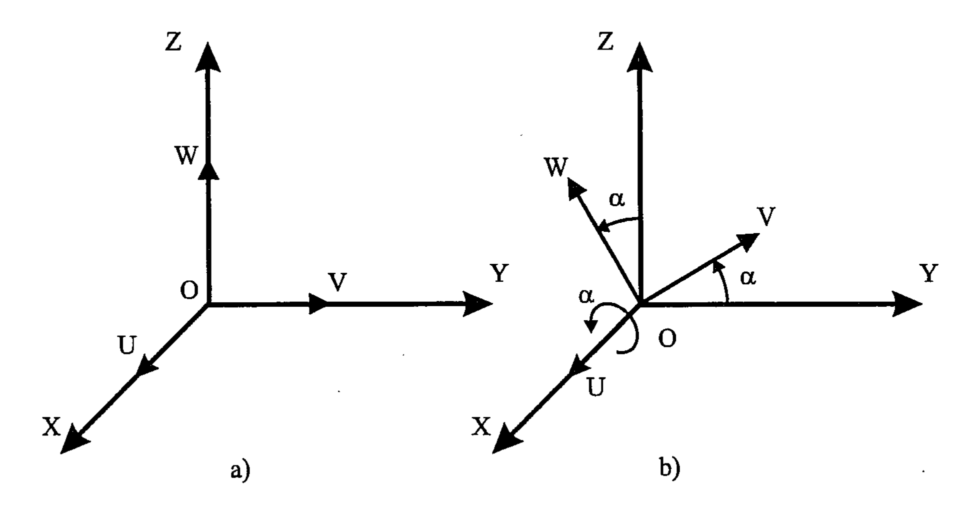


Figura 3.5. Sistema de referencia OXYZ y solidario al objeto OUVW.

vendrá representada mediante la matriz:

$$\mathbf{R}(\mathbf{y},\phi)=\begin{bmatrix}\cos\phi & 0 & \operatorname{sen}\phi\\ 0 & 1 & 0\\ -\operatorname{sen}\phi & 0 & \cos\phi\end{bmatrix} \qquad [3.\,9]$$

En la Figura 3.6-b, la orientación del sistema OUVW, con el eje OW coincidente con el eje OZ, vendrá representada mediante la matriz:

$$\mathbf{R}(\mathbf{z},\theta)=\begin{bmatrix}\cos\theta & -\operatorname{sen}\theta & 0\\ \operatorname{sen}\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix} \qquad [3.\,10]$$

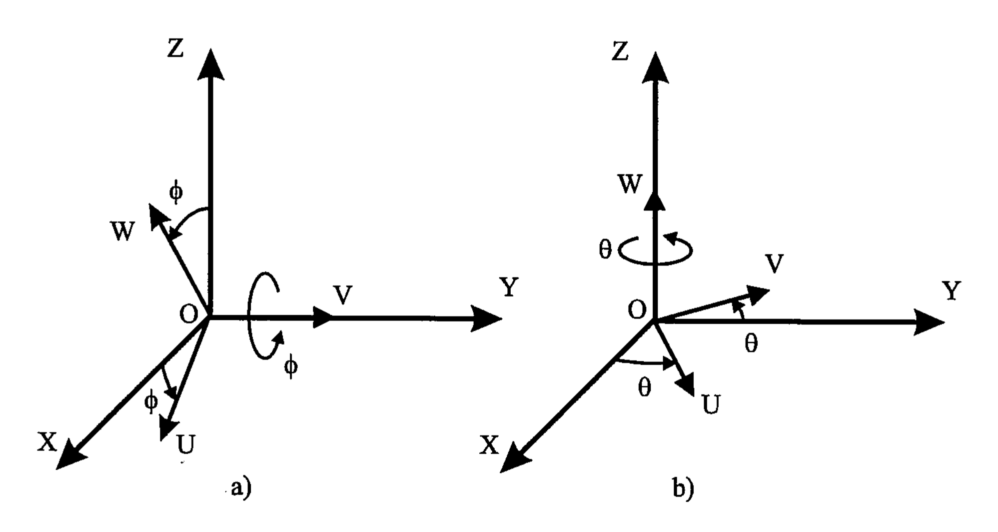


Figura 3.6. Rotación del sistema OUVW con respecto a los ejes OY y OZ.

Estas tres matrices, ecuaciones [3.8], [3.9] y [3.10], se denominan **matrices básicas de rotación** de un sistema espacial de tres dimensiones.

Composición de rotaciones

Las matrices de rotación pueden componerse para expresar la aplicación continua de varias rotaciones. Así, si al sistema OUVW se le aplica una rotación de ángulo α sobre OX, seguida de una rotación de ángulo ϕ sobre OY y de una rotación de ángulo θ sobre OZ, la rotación global puede expresarse como:

$$\mathbf{T}=\mathbf{R}(\mathbf{z},\theta)\mathbf{R}(\mathbf{y},\phi)\mathbf{R}(\mathbf{x},\alpha)=\begin{bmatrix}C\theta & -S\theta & 0\\ S\theta & C\theta & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}C\phi & 0 & S\phi\\ 0 & 1 & 0\\ -S\phi & 0 & C\phi\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & C\alpha & -S\alpha\\ 0 & S\alpha & C\alpha\end{bmatrix}=$$
$$=\begin{bmatrix}C\theta C\phi & -S\theta C\alpha+C\theta S\phi S\alpha & S\theta S\alpha+C\theta S\phi C\alpha\\ S\theta C\phi & C\theta C\alpha+S\theta S\phi S\alpha & -C\theta S\alpha+S\theta S\phi C\alpha\\ -S\phi & C\phi S\alpha & C\phi C\alpha\end{bmatrix} \quad [3.\ 11]$$

donde $C\theta$ expresa $\cos\theta$ y $S\theta$ expresa $\operatorname{sen}\theta$.

Es importante considerar el orden en que se realizan las rotaciones, pues el producto de matrices no es conmutativo. Así, si la rotación se hiciera primero un ángulo θ sobre OZ, seguida de una rotación de ángulo ϕ sobre OY, para finalizar con otra rotación de ángulo α sobre OX, la rotación global vendría expresada por:

$$\mathbf{T}=\mathbf{R}(\mathbf{x},\alpha)\mathbf{R}(\mathbf{y},\phi)\mathbf{R}(\mathbf{z},\theta)=\begin{bmatrix}1 & 0 & 0\\ 0 & C\alpha & -S\alpha\\ 0 & S\alpha & C\alpha\end{bmatrix}\begin{bmatrix}C\phi & 0 & S\phi\\ 0 & 1 & 0\\ -S\phi & 0 & C\phi\end{bmatrix}\begin{bmatrix}C\theta & -S\theta & 0\\ S\theta & C\theta & 0\\ 0 & 0 & 1\end{bmatrix}=$$
$$=\begin{bmatrix}C\phi C\theta & -C\phi S\theta & S\phi\\ S\alpha S\phi C\theta+C\alpha S\theta & -S\alpha S\phi S\theta+C\alpha C\theta & -S\alpha C\phi\\ -C\alpha S\phi C\theta+S\alpha S\theta & C\alpha S\phi S\theta+S\alpha C\theta & C\alpha C\phi\end{bmatrix} \quad [3.\ 12]$$

que como se aprecia difiere en gran medida de la anterior. Un estudio más detallado sobre la composición de rotaciones, aunque aplicado al caso más general de matrices de transformación homogénea, se puede encontrar en el epígrafe 3.3.5.

3.2.2. Ángulos de Euler

Para la representación de orientación en un espacio tridimensional mediante un matriz de rotación es necesario definir nueve elementos. Aunque la utilización de las matrices de rotación presente múltiples ventajas, como se verá en el siguiente epígrafe, existen otros métodos de definición de orientación que hacen únicamente uso de tres componentes para su descripción. Este es el caso de los llamados ángulos de Euler.

Todo sistema OUVW solidario al cuerpo cuya orientación se quiere describir, puede definirse con respecto al sistema OXYZ mediante tres ángulos: ϕ,θ,ψ, denominados **ángulos de Euler**. Girando sucesivamente el sistema OXYZ sobre unos ejes determinados de un triedro ortonormal los valores de ϕ,θ,ψ, se obtendrá el sistema OUVW. Es necesario, por tanto, conocer además de los valores de los

ángulos, cuáles son los ejes sobre los que se realizan los giros. Existen diversas posibilidades (24 formalmente definidas), siendo las tres más usuales las que se muestran a continuación:

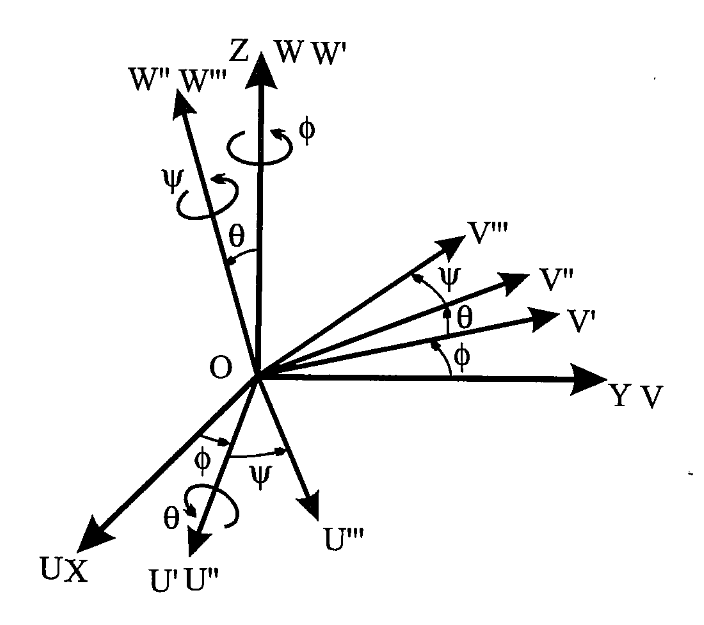


Figura 3.7. Ángulos de Euler ZXZ.

Ángulos de Euler ZXZ

Es una de las representaciones más habituales entre las que realizan los giros sobre ejes previamente girados. Se le suele asociar con los movimientos básicos de un giróscopo. Si se parte de los sistemas OXYZ y OUVW, inicialmente coincidentes, se puede colocar al sistema OUVW en cualquier orientación siguiendo los siguientes pasos. (Figura 3.7.)

1. Girar el sistema OUVW un ángulo ϕ con respecto al eje OZ, convirtiéndose así en el OU'V'W'.
2. Girar el sistema OU'V'W' un ángulo θ con respecto al eje OU', convirtiéndose así en el OU''V''W''.
3. Girar el sistema OU''V''W'' un ángulo ψ con respecto al eje OW'' convirtiéndose finalmente en el OU'''V'''W'''.

Es importante que estas operaciones se realicen en la secuencia especificada, pues las operaciones de giros consecutivos sobre ejes no son conmutativas.

Ángulos de Euler ZYZ

Es otra de las representaciones más habituales entre las que realizan los giros sobre ejes previamente girados. Sólo se diferencia del anterior en la elección del eje sobre el que se realiza el segundo giro. Si se

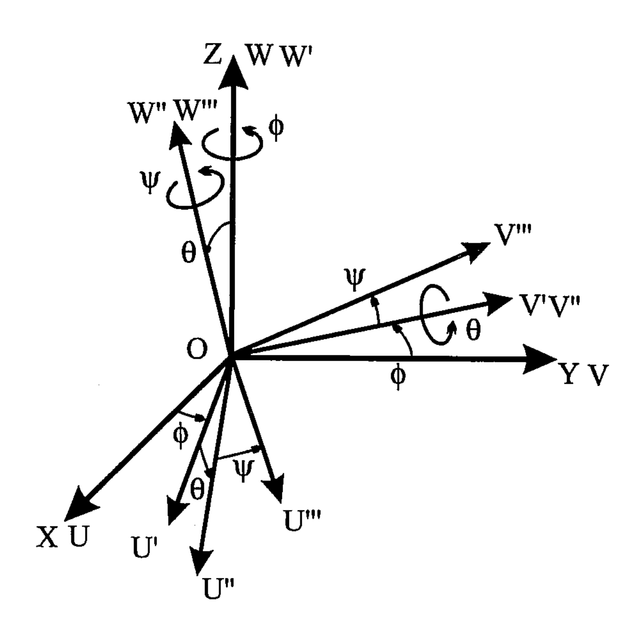


Figura 3.8. Ángulos de Euler ZYZ.

parte de los sistemas OXYZ y OUVW, inicialmente coincidentes, se puede colocar al sistema OUVW en cualquier orientación siguiendo los siguientes pasos. (Figura 3.8.)

1. Girar el sistema OUVW un ángulo ϕ con respecto al eje OZ, convirtiéndose así en el OU'V'W'.
2. Girar el sistema OU'V'W' un ángulo θ con respecto al eje OV', convirtiéndose así en el sistema OU''V''W''.
3. Girar el sistema OU''V''W'' un ángulo ψ con respecto al eje OW'', convirtiéndose finalmente en el OU'''V'''W'''.

Como antes, es preciso considerar que el orden de los giros no es conmutativo.

Roll, pitch and yaw (alabeo, cabeceo y guiñada)

Se trata de la representación utilizada generalmente en aeronáutica. Es también la más habitual de entre las que se aplican a los giros sobre los ejes del sistema fijo. Si se parte de los sistemas OXYZ y OUVW, al igual que en el caso anterior, se puede colocar al sistema OUVW en cualquier orientación siguiendo los siguientes pasos. (Figura 3.9.)

1. Girar el sistema OUVW un ángulo ψ con respecto al eje OX. Es el denominado Yaw o guiñada.
2. Girar el sistema OUVW un ángulo θ con respecto al eje OY. Es el denominado Pitch o cabeceo.
3. Girar el sistema OUVW un ángulo ϕ con respecto al eje OZ. Es el denominado Roll o alabeo.

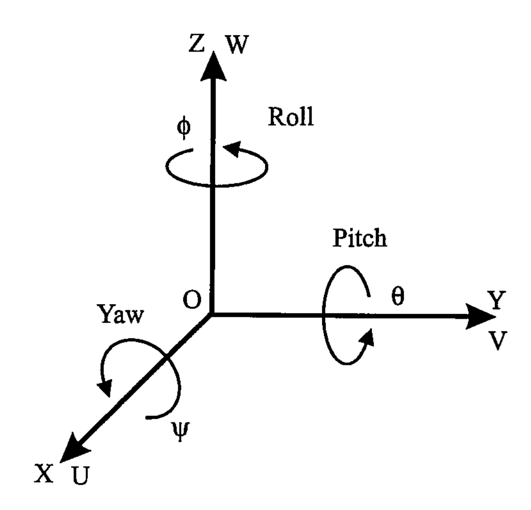


Figura 3.9. Ángulos de Euler: roll, pitch y yaw.

Al igual que en los casos anteriores, y en general siempre que se concatenan varios giros seguidos, es necesario considerar que no se trata de una transformación conmutativa, debiéndose seguir una secuencia determinada de aplicación de los mismos.

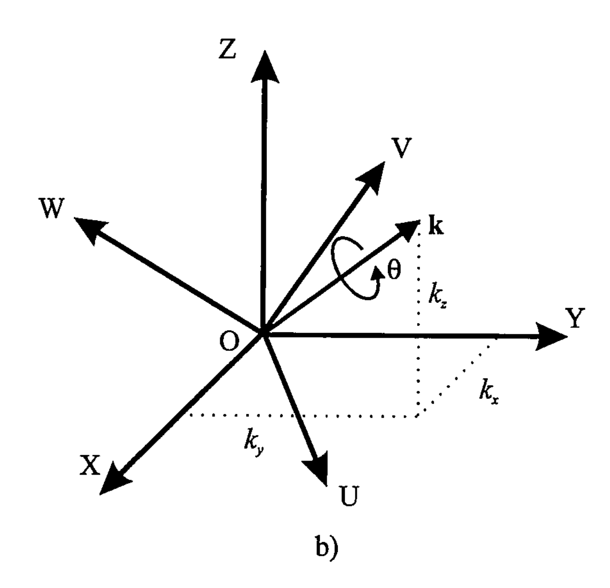


Figura 3.10. Representación de orientación por eje y ángulo de giro (par de rotación).

3.2.3. Par de rotación

La representación de la orientación de un sistema OUVW con respecto al sistema de referencia OXYZ también puede realizarse mediante la definición de un vector **k** (k_x,k_y.k_z) y un ángulo de giro θ, tal que el sistema OUVW corresponde al sistema OXYZ girado un ángulo θ sobre el eje **k** (Figura 3.10). El eje **k** ha de pasar por el origen O de ambos sistemas. Al par (**k**, θ) se le denomina par de rotación y se puede demostrar que es único.

Al igual que los ángulos de Euler, no se trata de un método que permita realizar una visualización sencilla de la orientación, salvo en casos muy concretos en los que el vector **k** coincida con algunos de los ejes coordenados del sistema OXYZ. La utilidad de este sistema se verá en epígrafes posteriores. Para la definición de orientación con este método, es necesario definir cuatro parámetros distintos: k_x,k_y.k_z y θ. Se puede representar como **Rot(k**,θ).

La aplicación de un par de rotación que rote un vector **p** un ángulo θ alrededor del eje **k** se realiza a través de la siguiente expresión:

$$\mathbf{Rot}(\mathbf{k},\theta)\,\mathbf{p} = \mathbf{p}\cos\theta - (\mathbf{k}\times\mathbf{p})\,\mathrm{sen}\theta + \mathbf{k}(\mathbf{k}\cdot\mathbf{p})(1-\cos\theta) \qquad [3.\ 13]$$

3.2.4. Cuaternios

Los cuaternios, definidos por Hamilton [HAMILTON-69], pueden ser utilizados como herramienta matemática de gran versatilidad computacional para trabajar con giros y orientaciones. En la bibliografía clásica sobre robótica suelen ser obviados o no tratados con el suficiente detalle, a pesar de ser empleados por algunos robots comerciales (ABB). Para comprender la verdadera utilidad de los cuaternios, es necesario analizar sus propiedades y ver la aplicación práctica de las mismas. Esto se realizará en un epígrafe posterior, exponiéndose aquí únicamente su definición

Un cuaternio Q está constituido por cuatro componentes (q_0,q_1,q_2,q_3) que representan las coordenadas del cuaternio en una base {**e,i,j,k**}. Es frecuente denominar parte escalar del cuaternio a la componente en **e**: q_0, y parte vectorial al resto de componentes. De modo que un cuaternio se puede representar como:

$$\mathrm{Q} = [q_0, q_1, q_2, q_3] = [\mathrm{s}, \mathbf{v}] \qquad [3.\ 14]$$

donde s representa la parte escalar, y **v** la parte vectorial.

Para la utilización de los cuaternios como metodología de representación de orientaciones se asocia el giro de un ángulo θ sobre el vector **k** al cuaternio definido por:

$$\mathrm{Q} = \mathbf{Rot}(\mathbf{k},\theta) = \left(\cos\frac{\theta}{2}, \mathbf{k}\ \mathrm{sen}\frac{\theta}{2}\right) \qquad [3.\ 15]$$

De esta asociación arbitraria y gracias a las propiedades de los cuaternios que más adelante se verán, se obtiene una importante herramienta analítica para el tratamiento de giros y cambios de orientación.

3.3. MATRICES DE TRANSFORMACIÓN HOMOGÉNEA

En los epígrafes anteriores se han estudiado distintos métodos de representar la posición o la orientación de un sólido en el espacio. Pero ninguno de estos métodos por sí solo permite una representación conjunta de la posición y de la orientación (localización). Para solventar este problema se introdujeron las denominadas coordenadas homogéneas.

3.3.1. Coordenadas y matrices homogéneas

La representación mediante coordenadas homogéneas de la localización de sólidos en un espacio n-dimensional se realiza a través de coordenadas de un espacio (n+1)-dimensional. Es decir, un espacio n-dimensional se encuentra representado en coordenadas homogéneas por (n+1) dimensiones, de tal forma que un vector $\mathbf{p}(x,y,z)$ vendrá representado por $\mathbf{p}(wx,wy,wz,w)$, donde w tiene un valor arbitrario y representa un factor de escala. De forma general, un vector $\mathbf{p} = a\mathbf{i} + b\mathbf{j} + c\mathbf{k}$, donde **i**, **j** y **k** son los vectores unitarios de los ejes OX, OY y OZ del sistema de referencia OXYZ, se representa en coordenadas homogéneas mediante el vector columna:

$$\mathbf{p} = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} aw \\ bw \\ cw \\ w \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \\ 1 \end{bmatrix} \qquad [3.16]$$

Por ejemplo, el vector $2\mathbf{i} + 3\mathbf{j} + 4\mathbf{k}$ se puede representar en coordenadas homogéneas como $[2,3,4,1]^T$, o como $[4,6,8,2]^T$ o también como $[-6,-9,-12,-3]^T$, etc. Los vectores nulo se representan como $[0,0,0,n]^T$, donde n es no-nulo. Los vectores de la forma $[a,b,c,0]^T$ sirven para representar direcciones, pues representan vectores de longitud infinita.

A partir de la definición de las coordenadas homogéneas surge inmediatamente el concepto de matriz de transformación homogénea. Se define como **matriz de transformación homogénea T** a una matriz de dimensión 4 x 4 que representa la transformación de un vector de coordenadas homogéneas de un sistema de coordenadas a otro.

$$\mathbf{T} = \begin{bmatrix} \mathbf{R}_{3x3} & \mathbf{p}_{3x1} \\ \mathbf{f}_{1x3} & \mathbf{w}_{1x1} \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ \text{Perspectiva} & \text{Escalado} \end{bmatrix} \qquad [3.17]$$

Se puede considerar que una matriz homogénea se haya compuesta por cuatro submatrices de distinto tamaño: una submatriz $\mathbf{R}_{3x3}$ que corresponde a una matriz de rotación; una submatriz $\mathbf{p}_{3x1}$ que corresponde al vector de traslación; una submatriz $\mathbf{f}_{1x3}$ que representa una transformación de perspectiva, y una submatriz $\mathbf{w}_{1x1}$ que representa un escalado global. En robótica generalmente sólo interesará conocer el valor de $\mathbf{R}_{3x3}$ y de $\mathbf{p}_{3x1}$, considerándose las componentes de $\mathbf{f}_{1x3}$ nulas y la de $\mathbf{w}_{1x1}$ la unidad, aunque más adelante se estudia su utilidad en otros campos. Al tratarse de una matriz 4 x 4, los vectores sobre los que se aplique deberán contar con 4 dimensiones, que serán las coordenadas homogéneas del vector tridimensional de que se trate.

3.3.2. Aplicación de las matrices homogéneas

Si como se ha mencionado, se considera la transformación de perspectiva nula y el escalado global unitario, la matriz homogénea **T** resultará ser de la siguiente forma:

$$\mathbf{T} = \begin{bmatrix} \mathbf{R}_{3x3} & \mathbf{p}_{3x1} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ 0 & 1 \end{bmatrix} \qquad [3.18]$$

que representa la orientación y posición de un sistema O'UVW rotado y trasladado con respecto al sistema de referencia OXYZ. Esta matriz sirve para conocer las coordenadas (r_x, r_y, r_z) del vector **r** en el sistema OXYZ a partir de sus coordenadas (r_u, r_v, r_w) en el sistema ~~OXYZ~~: O'UVW

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix} \qquad [3.19]$$

También se puede utilizar para expresar la rotación y traslación de un vector respecto de un sistema de referencia fijo OXYZ, de tal manera que un vector $\mathbf{r}_{xyz}$ rotado según $\mathbf{R}_{3x3}$ y trasladado según $\mathbf{P}_{3x1}$ se convierte en el vector $\mathbf{r'}_{xyz}$ dado por:

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} \qquad [3.20]$$

En resumen, una matriz de transformación homogénea se puede aplicar para:

1. Representar la posición y orientación de un sistema girado y trasladado O'UVW con respecto a un sistema fijo de referencia OXYZ, que es lo mismo que representar una rotación y traslación realizada sobre un sistema de referencia.
2. Transformar un vector expresado en coordenadas con respecto a un sistema O'UVW, a su expresión en coordenadas del sistema de referencia OXYZ.
3. Rotar y trasladar un vector con respecto a un sistema de referencia fijo OXYZ.

Se hace notar que se utilizan coordenadas homogéneas con factor de escalado la unidad, y que por tanto los vectores que intervienen en las transformaciones han de poseer cuatro componentes. Por comodidad, se elige el factor de escalado $w = 1$.

A continuación se va a analizar con detalle el empleo de las matrices homogéneas como herramienta para representar la localización de objetos en el espacio tridimensional, así como para realizar proyecciones y escalados.

Traslación

Supóngase que el sistema O'UVW únicamente se encuentra trasladado un vector $\mathbf{p} = p_x\mathbf{i} + p_y\mathbf{j} + p_z\mathbf{k}$ con respecto al sistema OXYZ. La matriz **T** entonces corresponderá a una matriz homogénea de traslación:

$$\mathbf{T}(\mathbf{p}) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 21]$$

que es la denominada matriz básica de traslación.

Un vector cualquiera **r**, representado en el sistema O'UVW por $\mathbf{r}_{uvw}$, tendrá como componentes del vector con respecto al sistema OXYZ:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_u \\ r_v \\ r_w \\ 1 \end{bmatrix} = \begin{bmatrix} r_u + p_x \\ r_v + p_y \\ r_w + p_z \\ 1 \end{bmatrix} \qquad [3.\ 22]$$

Y a su vez, un vector $\mathbf{r}_{x,y,z}$ desplazado según **T** tendrá como componentes $\mathbf{r}'_{x,y,z}$:

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} r_x + p_x \\ r_y + p_y \\ r_z + p_z \\ 1 \end{bmatrix} \qquad [3.\ 23]$$

Ejemplo 3.1.

Según la Figura 3.11 el sistema O'UVW está trasladado un vector **p**(6,-3,8) con respecto del sistema OXYZ. Calcular las coordenadas (r_x, r_y, r_z) del vector **r** cuyas coordenadas con respecto al sistema O'UVW son $\mathbf{r}_{uvw}$(-2,7,3).

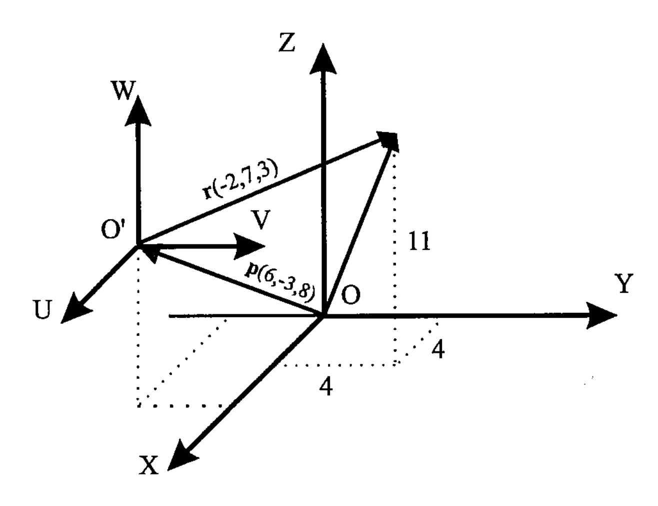


Figura 3.11. Figura del Ejemplo 3.1.

Aplicando la ecuación [3.22] se obtiene:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ 7 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \\ 11 \\ 1 \end{bmatrix}$$

Ejemplo 3.2.

Calcular el vector $\mathbf{r}'_{xyz}$ resultante de trasladar al vector $\mathbf{r}_{xyz}(4,4,11)$ según la transformación $\mathbf{T}(\mathbf{p})$ con $\mathbf{p}(6,-3,8)$ (ver Figura 3.12).

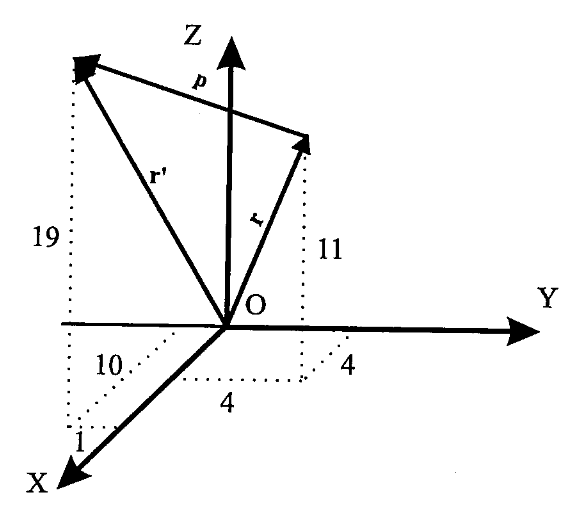


Figura 3.12. Figura del Ejemplo 3.2.

Aplicando la ecuación [3.23] se obtiene:

$$\begin{bmatrix} r'_x \\ r'_y \\ r'_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 6 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & 1 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 4 \\ 11 \\ 1 \end{bmatrix} = \begin{bmatrix} 10 \\ 1 \\ 19 \\ 1 \end{bmatrix}$$

Rotación

Supóngase ahora que el sistema O'UVW sólo se encuentra rotado con respecto al sistema OXYZ. La submatriz de rotación $\mathbf{R}_{3x3}$ será la que defina la rotación, y se corresponde al tipo matriz de rotación presentada en el epígrafe 3.2.1. De igual forma que se hacía allí, se pueden definir tres matrices homogéneas básicas de rotación según se realice ésta según cada uno de los tres ejes coordenados OX, OY y OZ del sistema de referencia OXYZ:

$$\mathbf{T}(\mathbf{x},\alpha)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\alpha & -\operatorname{sen}\alpha & 0\\ 0 & \operatorname{sen}\alpha & \cos\alpha & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [3.\ 24]$$

$$\mathbf{T}(\mathbf{y},\phi)=\begin{bmatrix}\cos\phi & 0 & \operatorname{sen}\phi & 0\\ 0 & 1 & 0 & 0\\ -\operatorname{sen}\phi & 0 & \cos\phi & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [3.\ 25]$$

$$\mathbf{T}(\mathbf{z},\theta)=\begin{bmatrix}\cos\theta & -\operatorname{sen}\theta & 0 & 0\\ \operatorname{sen}\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [3.\ 26]$$

Un vector cualquiera **r**, representado en el sistema girado O'UVW por $\mathbf{r}_{uvw}$, tendrá como componentes (r_x, r_y, r_z) en el sistema OXYZ las siguientes:

$$\begin{bmatrix}r_x\\ r_y\\ r_z\\ 1\end{bmatrix}=\mathbf{T}\begin{bmatrix}r_u\\ r_v\\ r_w\\ 1\end{bmatrix} \qquad [3.\ 27]$$

Y a su vez un vector $\mathbf{r}_{x,y,z}$ rotado según **T** vendrá expresado por $\mathbf{r}'_{x,y,z}$ según:

$$\begin{bmatrix}r'_x\\ r'_y\\ r'_z\\ 1\end{bmatrix}=\mathbf{T}\begin{bmatrix}r_x\\ r_y\\ r_z\\ 1\end{bmatrix} \qquad [3.\ 28]$$

Ejemplo 3.3.

Según la Figura 3.13, el sistema OUVW se encuentra girado -90° alrededor del eje OZ con respecto al sistema OXYZ. Calcular las coordenadas del vector $\mathbf{r}_{xyz}$ si $\mathbf{r}_{uvw} = [4,8,12]^T$.

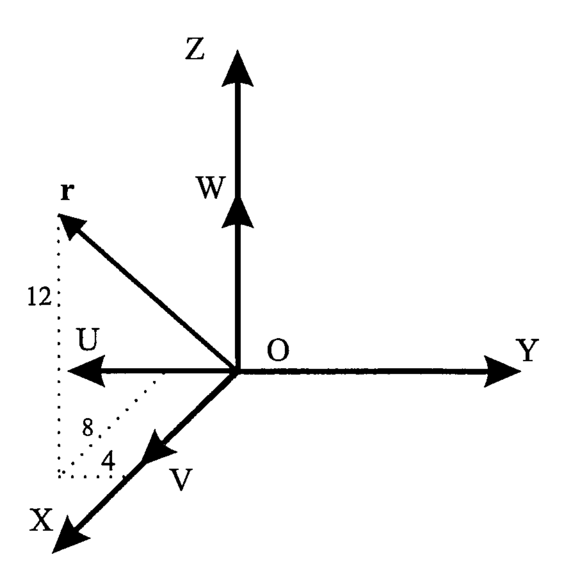


Figura 3.13. Figura del Ejemplo 3.3.

Aplicando la ecuación [3.27] y la matriz de la expresión [3.26] se tendrá:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 8 \\ 12 \\ 1 \end{bmatrix} = \begin{bmatrix} 8 \\ -4 \\ 12 \\ 1 \end{bmatrix}$$

Traslación junto con rotación

La principal ventaja de las matrices homogéneas reside en su capacidad de representación conjunta de posición y orientación. Esta representación se realiza utilizando al mismo tiempo la matriz de rotación $\mathbf{R}_{3x3}$ y el vector de traslación $\mathbf{p}_{3x1}$ en una misma matriz de transformación homogénea. Es por tanto la aplicación conjunta de lo visto en los dos apartados anteriores.

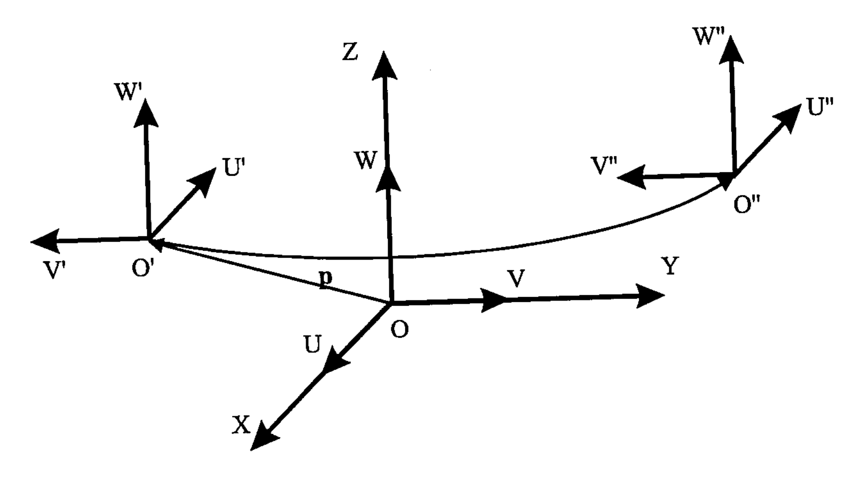


Figura 3.14. Distintos sistemas finales según el orden de las transformaciones.

La traslación y la rotación son transformaciones que se realizan en relación a un sistema de referencia. Por lo tanto, si se quiere expresar la posición y orientación de un sistema O'UVW, originalmente coincidente con el de referencia y que ha sido rotado y trasladado según éste, habrá que tener en cuenta si primero se ha realizado la rotación y después la traslación o viceversa, pues se trata de transformaciones espaciales no conmutativas. En la Figura 3.14 se demuestra esta no conmutatividad de forma gráfica. Se parte de un sistema OUVW coincidente con OXYZ al que se va a aplicar una traslación según un vector $\mathbf{p}_{x,y,z}$ y una rotación de 180° alrededor del eje OZ. Si primero se rota y después se traslada se obtiene un sistema final O'U'V'W'. En cambio, si primero se traslada y después se rota se obtiene otro sistema final O"U"V"W", que representa una localización totalmente distinta a la del sistema final anterior. Se tendrá, por tanto, matrices homogéneas distintas según se realice una traslación seguida de rotación o una rotación seguida de traslación.

Rotación seguida de traslación

Para el caso de realizar primero una rotación sobre uno de los ejes coordenados del sistema OXYZ seguida de una traslación, las matrices homogéneas serán las que a continuación se expresan:

Rotación de un ángulo α sobre el eje OX seguido de un traslación de vector $\mathbf{p}_{x,y,z}$:

$$\mathbf{T}((\mathbf{x},\alpha),\mathbf{p}) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & \cos\alpha & -\operatorname{sen}\alpha & p_y \\ 0 & \operatorname{sen}\alpha & \cos\alpha & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.29]$$

Rotación de un ángulo ϕ sobre el eje OY seguido de una traslación de vector $\mathbf{p}_{x,y,z}$:

$$\mathbf{T}((\mathbf{y},\phi),\mathbf{p}) = \begin{bmatrix} \cos\phi & 0 & \operatorname{sen}\phi & p_x \\ 0 & 1 & 0 & p_y \\ -\operatorname{sen}\phi & 0 & \cos\phi & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 30]$$

Rotación de un ángulo θ sobre el eje OZ seguido de una traslación de vector $\mathbf{p}_{x,y,z}$:

$$\mathbf{T}((\mathbf{z},\theta),\mathbf{p}) = \begin{bmatrix} \cos\theta & -\operatorname{sen}\theta & 0 & p_x \\ \operatorname{sen}\theta & \cos\theta & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 31]$$

Traslación seguida de rotación

Para el caso de realizar primero una traslación seguida de una rotación sobre los ejes coordenados del sistema OXYZ, las matrices homogéneas resultantes son las siguientes:

Traslación de vector $\mathbf{p}_{x,y,z}$ seguida de rotación de un ángulo α sobre el eje OX.

$$\mathbf{T}(\mathbf{p},(\mathbf{x},\alpha)) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & \cos\alpha & -\operatorname{sen}\alpha & p_y\cos\alpha - p_z\operatorname{sen}\alpha \\ 0 & \operatorname{sen}\alpha & \cos\alpha & p_y\operatorname{sen}\alpha + p_z\cos\alpha \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 32]$$

Traslación de vector $\mathbf{p}_{x,y,z}$ seguida de rotación de un ángulo ϕ sobre el eje OY.

$$\mathbf{T}(\mathbf{p},(\mathbf{y},\phi)) = \begin{bmatrix} \cos\phi & 0 & \operatorname{sen}\phi & p_x\cos\phi + p_z\operatorname{sen}\phi \\ 0 & 1 & 0 & p_y \\ -\operatorname{sen}\phi & 0 & \cos\phi & p_z\cos\phi - p_x\operatorname{sen}\phi \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 33]$$

Traslación de vector $\mathbf{p}_{x,y,z}$ seguida de rotación de un ángulo θ sobre el eje OZ.

$$\mathbf{T}(\mathbf{p},(\mathbf{z},\theta)) = \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 & p_x\cos\theta - p_y\text{sen}\theta \\ \text{sen}\theta & \cos\theta & 0 & p_x\text{sen}\theta + p_y\cos\theta \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [3.\ 34]$$

Ejemplo 3.4.

Un sistema OUVW ha sido girado 90° alrededor del eje OX y posteriormente trasladado un vector $\mathbf{p}$(8,-4,12) con respecto al sistema OXYZ (Figura 3.15). Calcular las coordenadas (r_x, r_y, r_z) del vector $\mathbf{r}$ con coordenadas $\mathbf{r}_{uvw}$(-3,4,-11).

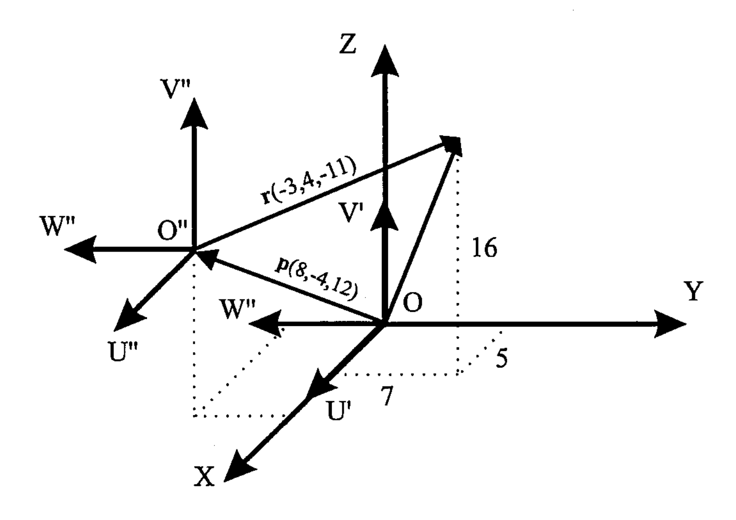


Figura 3.15. Sistemas de referencia del Ejemplo 3.4.

Utilizando la matriz de la expresión [3.29] se tendrá:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 0 & -1 & -4 \\ 0 & 1 & 0 & 12 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -3 \\ 4 \\ -11 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \\ 16 \\ 1 \end{bmatrix}$$

Ejemplo 3.5.

Un sistema OUVW ha sido trasladado un vector **p**(8,-4,12) con respecto al sistema OXYZ y girado 90° alrededor del eje OX (Figura 3.16). Calcular las coordenadas (r_x, r_y, r_z) del vector **r** de coordenadas r_{uvw} (-3,4,-11).

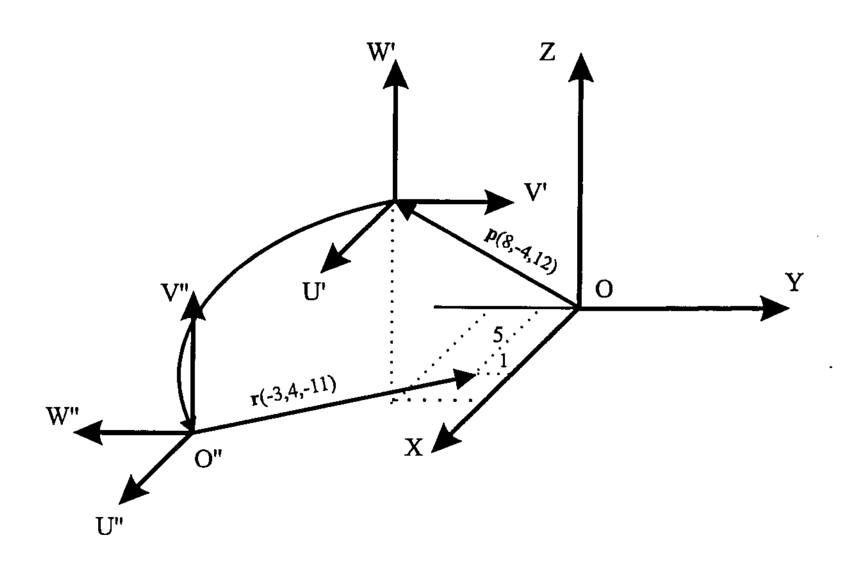


Figura 3.16. Situación de los sistemas de referencia del Ejemplo 3.5.

Utilizando la matriz de la expresión [3.32] se obtiene:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 8 \\ 0 & 0 & -1 & -12 \\ 0 & 1 & 0 & -4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -3 \\ 4 \\ -11 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ -1 \\ 0 \\ 1 \end{bmatrix}$$

Perspectiva y escalado

Las matrices homogéneas también se pueden aplicar para la realización de un escalado de las componentes de un vector. Bastará utilizar una matriz **T** del tipo:

$$\mathbf{T}=\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [3.35]$$

Cualquier vector $\mathbf{r}(x,y,z)$ puede ser transformado en el vector $\mathbf{r}(ax,by,cz)$. También se puede realizar un escalado global de las tres componentes mediante la matriz:

$$\mathbf{T}=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & s \end{bmatrix} \quad [3.36]$$

a través de la cual, utilizando la definición de coordenadas homogéneas, cualquier vector $\mathbf{r}(x,y,z)$ puede ser transformado en un vector $\mathbf{r}(x/s,y/s,z/s)$.

Una aplicación más de las matrices homogéneas es la transformación de perspectiva. Supóngase una lente situada sobre el plano OXZ con distancia focal f situada sobre el eje OY (Figura 3.17). Se puede comprobar que el punto $r(x,y,z)$ se ve en el plano de la lente como un punto $r'(x',y',z')$ cuyas coordenadas vienen dadas por la siguiente expresión:

$$x'=\frac{x}{1-\frac{y}{f}} \qquad y'=0 \qquad z'=\frac{z}{1-\frac{y}{f}} \quad [3.37]$$

Esta transformación, denominada de perspectiva, puede realizarse a través de una matriz homogénea del tipo [FU-88]:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1/f & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad [3.38]$$

Para las aplicaciones en robótica de las matrices homogéneas, se supone que no existe ninguna transformación de perspectiva y que el escalado es siempre unitario.

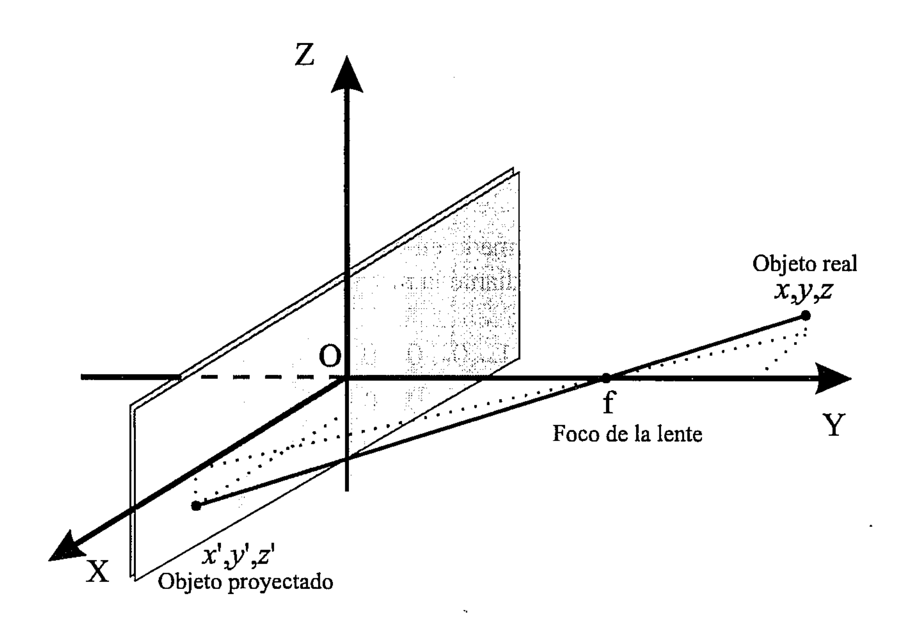


Figura 3.17. Transformación de perspectiva.

3.3.3. Significado geométrico de las matrices homogéneas

Como ya se ha descrito, una matriz homogénea sirve para transformar un vector expresado en coordenadas homogéneas con respecto a un sistema O'UVW, a su expresión en las coordenadas del sistema de referencia OXYZ. También se puede utilizar para rotar y girar un vector referido a un sistema de referencia fijo, y en definitiva sirve para expresar la orientación y posición de un sistema de referencia O'UVW con respecto a otro fijo OXYZ.

La matriz **T** de transformación se suele escribir de la siguiente forma:

$$\mathbf{T} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & \mathrm{p}_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 39]$$

donde **n**,**o**,**a** es una terna ortonormal que representa la orientación y **p** es un vector que representa la posición.

Si se considera un vector $\mathbf{r}_{uvw} = [0,0,0,1]^T$, es decir, el origen del sistema O'UVW, la aplicación de la matriz **T** que representa la transformación (traslación + rotación) de O'UVW con respecto a OXYZ, se obtiene $\mathbf{r}_{xyz}$:

$$\mathbf{r}_{xyz} = \begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} \qquad [3.40]$$

que coincide con el vector columna **p** de **T**. Por tanto, este vector columna representa la posición del origen de O'UVW con respecto del sistema OXYZ.

Si, de igual manera, se considera el vector de coordenadas homogéneas $[1,0,0,1]^T$ con respecto del sistema OUVW, es decir, el vector director del eje coordenado O'U del sistema O'UVW, y suponiendo el vector **p** de traslación nulo, se tendrá:

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} n_x \\ n_y \\ n_z \\ 1 \end{bmatrix} \qquad [3.41]$$

Es decir, el vector columna **n** representa las coordenadas del eje O'U del sistema O'UVW con respecto del sistema OXYZ. De igual forma, si se realiza la transformación de los vectores $[0,1,0,1]^T$ y $[0,0,1,1]^T$ referidos al sistema O'UVW, se obtiene que el vector columna **o** representa las coordenadas del eje O'V del sistema O'UVW con respecto del sistema OXYZ, y que el vector columna **a** representa las coordenadas del eje O'W del sistema O'UVW con respecto del sistema OXYZ.

Consecuentemente, los vectores **n, o** y **a** definen una terna ortonormal a derechas, lo que significa que:

$$\begin{gathered} \|\mathbf{n}\| = \|\mathbf{o}\| = \|\mathbf{a}\| = 1 \\ \mathbf{n} \times \mathbf{o} = \mathbf{a} \end{gathered} \qquad [3.42]$$

Como ya se vio cuando se explicó la matriz de rotación, la submatriz de rotación [**n,o,a**] corresponde a una matriz ortonormal, que cumple que:

$$\begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} \end{bmatrix}^T \qquad [3.43]$$

La matriz inversa de la matriz homogénea de transformación **T** es fácilmente obtenible, y corresponde a la siguiente expresión:

$$\mathbf{T}^{-1} = \begin{bmatrix} n_x & n_y & n_z & -\mathbf{n}^T\mathbf{p} \\ o_x & o_y & o_z & -\mathbf{o}^T\mathbf{p} \\ a_x & a_y & a_z & -\mathbf{a}^T\mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.44]$$

Si se tiene la relación $\mathbf{r}_{xyz} = \mathbf{T}\,\mathbf{r}_{uvw}$ y se multiplica en ambos miembros por $\mathbf{T}^{-1}$, se tiene:

$$\mathbf{T}^{-1}\mathbf{r}_{xyz} = \mathbf{r}_{uvw} \qquad [3.45]$$

por lo que, realizando el mismo proceso que se hizo anteriormente, se deduce que los vectores fila de la submatriz de rotación de la matriz **T** (vectores columna de la submatriz de rotación de $\mathbf{T}^{-1}$), representan los ejes principales del sistema de coordenadas de referencia OXYZ con respecto a OUVW. Es decir, los vectores fila de la matriz [**n o a**] representan otra terna ortonormal a derechas.

3.3.4. Composición de matrices homogéneas

Anteriormente se ha mencionado que una matriz de transformación homogénea sirve, entre otras cosas, para representar el giro y la traslación realizados sobre un sistema de referencia. Esta utilidad de las matrices homogéneas cobra aún más importancia cuando se componen las matrices homogéneas para describir diversos giros y traslaciones consecutivos sobre un sistema de referencia determinado.

De esta forma, una transformación compleja podrá descomponerse en la aplicación consecutiva de transformaciones simples (giros básicos y traslaciones).

Por ejemplo, una matriz que representa un giro de un ángulo α sobre el eje OX, seguido de un giro de ángulo ϕ sobre el eje OY y de un giro de un ángulo θ sobre el eje OZ, puede obtenerse por la composición de las matrices básicas de rotación:

$$\mathbf{T} = \mathbf{T}(\mathbf{z},\theta)\,\mathbf{T}(\mathbf{y},\phi)\,\mathbf{T}(\mathbf{x},\alpha) = \begin{bmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi & 0 & S\phi & 0 \\ 0 & 1 & 0 & 0 \\ -S\phi & 0 & C\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} C\phi C\theta & -S\theta C\alpha + C\theta S\phi S\alpha & S\theta S\alpha + C\theta S\phi C\alpha & 0 \\ S\theta C\phi & C\theta C\alpha + S\theta S\phi S\alpha & -C\theta S\alpha + S\theta S\phi C\alpha & 0 \\ -S\phi & C\phi S\alpha & C\alpha C\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.46]$$

Debido a que el producto de matrices no es conmutativo, tampoco lo es la composición de transformaciones. Si se invierte el orden de aplicación de las transformaciones, el resultado es, lógicamente, distinto:

$$\mathbf{T}=\mathbf{T}(\mathbf{x},\alpha)\,\mathbf{T}(\mathbf{y},\phi)\,\mathbf{T}(\mathbf{z},\theta)=\begin{bmatrix}1&0&0&0\\0&C\alpha&-S\alpha&0\\0&S\alpha&C\alpha&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}C\phi&0&S\phi&0\\0&1&0&0\\-S\phi&0&C\phi&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}C\theta&-S\theta&0&0\\S\theta&C\theta&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}= \quad [3.47]$$

$$=\begin{bmatrix}C\phi C\theta&-C\phi S\theta&S\phi&0\\S\alpha S\phi C\theta+C\alpha S\theta&-S\alpha S\phi S\theta+C\alpha C\theta&-S\alpha C\phi&0\\-C\alpha S\phi C\theta+S\alpha S\theta&C\alpha S\phi S\theta+S\alpha C\theta&C\alpha C\phi&0\\0&0&0&1\end{bmatrix}$$

Ejemplo 3.6.

Se quiere obtener la matriz de transformación que representa al sistema O'UVW obtenido a partir del sistema OXYZ mediante un giro de ángulo -90° alrededor del eje OX, de una traslación de vector $p_{xyz}(5,5,10)$ y un giro de 90° sobre el eje OZ.

Para ello bastará multiplicar en el orden adecuado las diferentes matrices básicas:

$$\mathbf{T}=\mathbf{T}(\mathbf{z},90^{\circ})\,\mathbf{T}(\mathbf{p})\,\mathbf{T}(\mathbf{x},\text{-}90^{\circ})=\begin{bmatrix}0&-1&0&0\\1&0&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&5\\0&1&0&5\\0&0&1&10\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&-1&0&0\\0&0&0&1\end{bmatrix}=\begin{bmatrix}0&0&-1&-5\\1&0&0&5\\0&-1&0&10\\0&0&0&1\end{bmatrix}$$

En los ejemplos vistos anteriormente los ejes sobre los que se realizaban las operaciones correspondían al sistema fijo de referencia OXYZ. También es posible componer matrices de transformación de manera que las operaciones estén referidas en todo momento al sistema que esté moviéndose. Para ello bastará unicamente con ir concatenando matrices en orden inverso. Por ejemplo, en la siguiente ecuación :

$$\mathbf{T} = \mathbf{T}(\mathbf{x},\alpha)\,\mathbf{T}(\mathbf{v},\phi)\,\mathbf{T}(\mathbf{w},\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi & 0 & S\phi & 0 \\ 0 & 1 & 0 & 0 \\ -S\phi & 0 & C\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} C\phi C\theta & -C\phi S\theta & S\phi & 0 \\ S\alpha S\phi C\theta + C\alpha S\theta & -S\alpha S\phi S\theta + C\alpha C\theta & -S\alpha C\phi & 0 \\ -C\theta C\alpha S\phi + S\alpha S\theta & C\alpha S\phi S\theta + S\alpha C\theta & C\alpha C\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.48]$$

se muestra una matriz que representa un giro de ángulo α sobre el eje OX del sistema fijo OXYZ, seguido de un giro de valor ϕ sobre el eje OV y un giro de ángulo θ sobre el eje OW del sistema en movimiento OUVW. Comparar con la matriz de la ecuación [3.46], que representa las mismas transformaciones referidas a los ejes de un sistema OXYZ fijo de referencia.

Ejemplo 3.7.

Obtener la matriz de transformación que representa las siguientes transformaciones sobre un sistena OXYZ fijo de referencia: traslación de un vector $\mathbf{p}_{xyz}(-3,10,10)$; giro de -90° sobre el eje O'U del sistema trasladado y giro de 90° sobre el eje O'V del sistema girado.

Se escogen las matrices básicas correspondientes y se componen en orden inverso al ejemplo anterior.

$$\mathbf{T} = \mathbf{T}(\mathbf{p})\,\mathbf{T}(\mathbf{u},-90^\circ)\,\mathbf{T}(\mathbf{v},90^\circ) = \begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 10 \\ 0 & 0 & 1 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & -3 \\ -1 & 0 & 0 & 10 \\ 0 & -1 & 0 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

De forma general, a la hora de componer diversas transformaciones mediante matrices homogéneas, se han de tener en cuenta los siguientes criterios:

1. Si el sistema fijo OXYZ y el sistema transformado O'UVW son coincidentes, la matriz homogénea de transformación será la matriz 4 x 4 identidad, $\mathbf{I}_4$.
2. Si el sistema O'UVW se obtiene mediante rotaciónes y traslaciones definidas con respecto al sistema fijo OXYZ, la matriz homogénea que representa cada transformación se deberá premultiplicar sobre las matrices de las transformaciones previas.

3. Si el sistema O'UVW se obtiene mediante rotaciones y traslaciones definidas con respecto al sistema móvil, la matriz homogénea que representa cada transformación se deberá postmultiplicar sobre las matrices de las transformaciones previas.

Siguiendo estas indicaciones, cualquier composición de matrices homogéneas puede estudiarse como si se realiza cada transformación con respecto al sistema fijo o se realiza cada transformación con respecto al sistema móvil.

Por ejemplo, la transformación:

$$\mathbf{T} = \mathbf{T}(\mathbf{x},\alpha)\,\mathbf{T}(\mathbf{z},\theta)\,\mathbf{T}(\mathbf{y},\phi) \qquad [3.\ 49]$$

puede verse como una rotación de ϕ sobre OY, seguida de una rotación θ sobre OZ y de una rotación α sobre OX del sistema fijo. O también puede verse cómo una rotación α sobre el eje OU, seguida de una rotación θ sobre el eje OW y de una rotación ϕ sobre el eje OV del sistema que está siendo transformado.

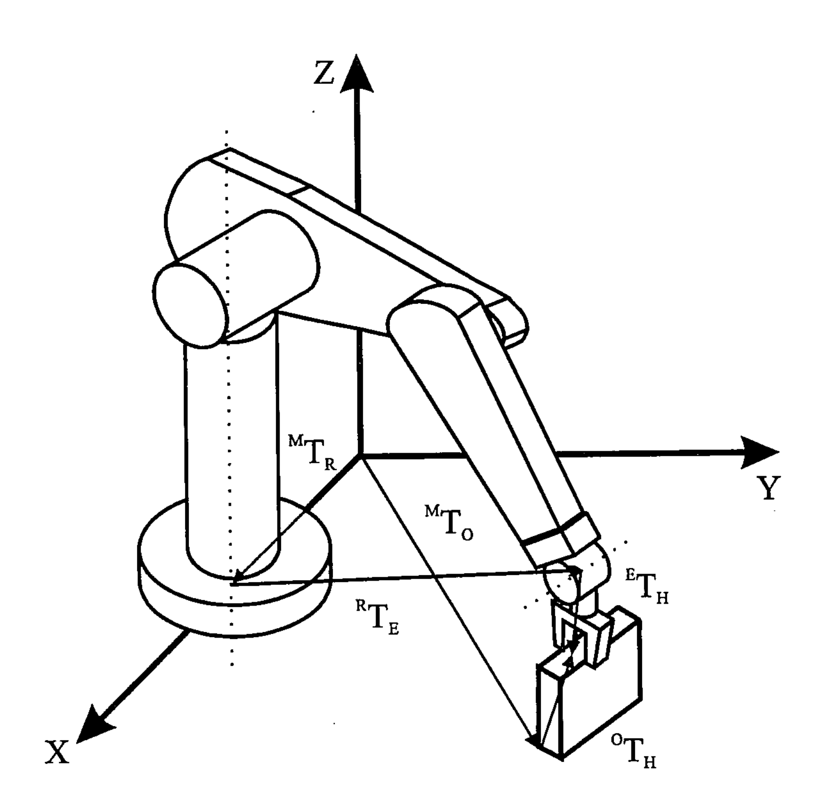


Figura 3.18. Ejemplo de aplicación de diversas transformaciones para localizar un objeto.

3.3.5. Gráficos de transformación

Es frecuente encontrar situaciones en las que la localización espacial de un objeto o de su sistema de referencia asociado, pueda realizarse a través de la composición de diversas transformaciones distintas. En la Figura 3.18 se tiene un manipulador cuya base está referida al sistema del mundo OXYZ mediante la transformación ${}^{M}\mathbf{T}_{R}$. A su vez, para pasar de la base del manipulador a su extremo se utiliza la transformación ${}^{R}\mathbf{T}_{E}$.

El extremo de la herramienta está referido con respecto al extremo del manipulador por la transformación ${}^{E}\mathbf{T}_{H}$. A su vez, un objeto está referido con respecto al sistema OXYZ mediante la transformación ${}^{M}\mathbf{T}_{O}$, y por último, el extremo de la herramienta está referido con respecto al objeto a través de la transformación ${}^{O}\mathbf{T}_{H}$.

Se observa que el final de la herramienta puede ser referido con respecto al sistema OXYZ de dos maneras distintas: a través del manipulador y a través del objeto. De tal manera que se puede escribir:

$${}^{M}\mathbf{T}_{R}\ {}^{R}\mathbf{T}_{E}\ {}^{E}\mathbf{T}_{H} = {}^{M}\mathbf{T}_{O}\ {}^{O}\mathbf{T}_{H} \qquad [3.50]$$

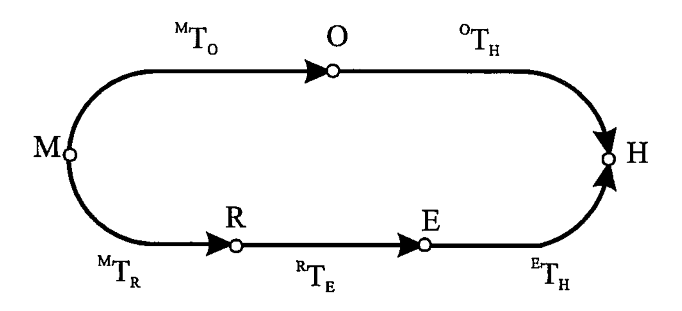


Figura 3.19. Gráfico de transformación.

Esta relación se puede representar mediante un gráfico de transformación como el de la Figura 3.19 [PAUL-81]. De tal manera que si se quiere obtener la relación entre el objeto y la herramienta bastará multiplicar ambos miembros de la ecuación anterior por ${}^{M}\mathbf{T}_{O}^{-1}$ obteniéndose:

$$\left({}^{M}\mathbf{T}_{O}\right)^{-1}\ {}^{M}\mathbf{T}_{R}\ {}^{R}\mathbf{T}_{E}\ {}^{E}\mathbf{T}_{H} = {}^{O}\mathbf{T}_{H} \qquad [3.51]$$

Cualquier otra relación puede ser obtenida fácilmente a partir del gráfico. Para ello se irá desde el objeto inicial al final multiplicando las matrices de transformación correspondiente a los arcos del gráfico, y considerando que de recorrerse éstos en el sentido inverso a las flechas deberá utilizarse una matriz inversa. Así la relación entre la base del robot y el objeto vendrá dada por:

$${}^{R}\mathbf{T}_{O} = {}^{R}\mathbf{T}_{E}\ {}^{E}\mathbf{T}_{H}\left({}^{O}\mathbf{T}_{H}\right)^{-1}$$

o bien por:

$$^{R}\mathbf{T}_{O} = \left(^{M}\mathbf{T}_{R} \right)^{-1} \; ^{M}\mathbf{T}_{O}$$

3.4. APLICACIÓN DE LOS CUATERNIOS

En el epígrafe 3.2.4 se definió de forma genérica lo que es un cuaternio. A continuación se describe el álgebra de cuaternios y las aplicaciones en las que la utilización de cuaternios supone una ventaja sustancial sobre otros métodos de descripción espacial.

3.4.1. Álgebra de cuaternios

Un cuaternio está formado por cuatro componentes (q_0, q_1, q_2, q_3) que representan las coordenadas del cuaternio en una base {**e**,**i**,**j**,**k**}.

$$Q = q_0\mathbf{e} + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} = (\mathrm{s}, \mathbf{v}) \qquad [3.\ 52]$$

Sobre los elementos de la base se define una ley de composición interna ∘ (producto) según se muestra en la Tabla 3.1. De este modo los cuaternios forman un grupo cíclico de orden cuatro. A continuación se describen algunas propiedades útiles de los cuaternios a la hora de su utilización para realizar transformaciones.

Tabla 3.1. Ley de composición interna de los cuaternios

∘	**e**	**i**	**j**	**k**
e	**e**	**i**	**j**	**k**
i	**i**	**-e**	**k**	**-j**
j	**j**	**-k**	**-e**	**i**
k	**k**	**j**	**-i**	**-e**

Cuaternio conjugado

A todo cuaternio Q se le puede asociar su conjugado Q^*, en el que se mantiene el signo de la parte escalar y se invierte el de la vectorial.

$$Q^* = [q_0, -q_1, -q_2, -q_3] = (s, -\mathbf{v}) \quad [3.53]$$

Operaciones algebráicas

Se definen tres operaciones algebráicas sobre los cuaternios: producto, suma y producto con un escalar.

El producto de dos cuaternios Q_1 y Q_2, que va a ser muy útil para la composición de transformaciones, puede decucirse de la Tabla 3.1., y viene dado por:

$$Q_3 = Q_1 \circ Q_2 = (s_1, \mathbf{v}_1) \circ (s_2, \mathbf{v}_2) = (s_1 s_2 - \mathbf{v}_1 \mathbf{v}_2, \mathbf{v}_1 \times \mathbf{v}_2 + s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1) \quad [3.54]$$

Se observa que no se trata de un producto conmutativo. Si se expresa componente a componente:

$$\begin{aligned} q_{30} &= q_{10} q_{20} - (q_{11} q_{21} + q_{12} q_{22} + q_{13} q_{23}) \\ q_{31} &= q_{10} q_{21} + q_{11} q_{20} + q_{12} q_{23} - q_{13} q_{22} \\ q_{32} &= q_{10} q_{22} + q_{12} q_{20} + q_{13} q_{21} - q_{11} q_{23} \\ q_{33} &= q_{10} q_{23} + q_{13} q_{20} + q_{11} q_{22} - q_{12} q_{21} \end{aligned} \quad [3.55]$$

La suma de dos cuaternios Q_1 y Q_2 se define como:

$$Q_3 = Q_1 + Q_2 = (s_1, \mathbf{v}_1) + (s_2, \mathbf{v}_2) = (s_1 + s_2, \mathbf{v}_1 + \mathbf{v}_2) \quad [3.56]$$

mientras que el producto por un escalar es:

$$Q_3 = a\,Q_2 = a(s_2, \mathbf{v}_2) = (a\,s_2, a\,\mathbf{v}_2) \quad [3.57]$$

El producto de cuaternios es por tanto asociativo aunque no conmutativo.

Norma e inverso

Según la definición del cuaternio conjugado y la del producto de cuaternios, se deduce que:

$$Q \circ Q^* = (q_0^2 + q_1^2 + q_2^2 + q_3^2)\mathbf{e} \quad [3.58]$$

Al número real $(q_0^2 + q_1^2 + q_2^2 + q_3^2)^{1/2}$ se le denomina norma de Q y se representa por $\|Q\|$. El inverso de un cuaternio puede hallarse mediante la expresión:

$$Q^{-1} = \frac{Q^*}{\|Q\|} \quad [3.59]$$

siempre y cuando se trate de un cuaternio no nulo.

3.4.2. Utilización de los cuaternios

Las propiedades expuestas propician el uso de los cuaternios para la representación y composición de rotaciones. Para ello, primeramente se define aquel cuaternio que represente un giro de valor θ sobre un eje **k** como:

$$Q = \mathrm{Rot}(\mathbf{k},\theta) = \left(\cos\frac{\theta}{2}, \mathbf{k}\,\mathrm{sen}\frac{\theta}{2}\right) \qquad [3.\,60]$$

Ejemplo 3.8.

Obtener el cuaternio que representa una rotación de 90° sobre el eje **k**(3,-2,1).

Aplicando la ecuación [3.60]:

$$Q = \mathrm{Rot}(\mathbf{k},90^\circ) = \left(\frac{\sqrt{2}}{2}, 3\frac{\sqrt{2}}{2}, \text{-}2\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$$

En segundo lugar, la aplicación de la rotación expresada por el cuaternio Q a un vector **r**, vendrá definida por el producto:

$$Q \circ (0,\mathbf{r}) \circ Q^* \qquad [3.\,61]$$

Ejemplo 3.9.

Obtener el vector **r'** resultante de aplicar la misma rotación del Ejemplo 3.8. Rot(**k**,90°) donde **k**(3,-2,1), sobre el vector **r**(5,2,-6).

Aplicando la ecuación [3.61] se obtiene el vector **r'**:

$$(0, \mathbf{r}') = \left(\frac{\sqrt{2}}{2}, 3\frac{\sqrt{2}}{2}, -2\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right) \circ (0,5,2,-6) \circ \left(\frac{\sqrt{2}}{2}, -3\frac{\sqrt{2}}{2}, 2\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right)$$

La composición de rotaciones con cuaternios resulta tan sencilla como multiplicar cuaternios entre sí. De tal forma que el resultado de rotar según el cuaternio Q_1, para posteriormente rotar según Q_2, es el mismo que el de rotar según Q_3, obtenido por la expresión:

$$Q_3 = Q_2 \circ Q_1 \qquad [3.62]$$

Es importante tener en cuenta el orden de multiplicación, pues como se ha mencionado, el producto de cuaternios no es conmutativo.

En el caso de componer rotaciones con traslaciones se procede como sigue: el resultado de aplicar una traslación de vector **p** seguida de una rotación Q al sistema OXYZ, es un nuevo sistema OUVW, tal que las coordenadas de un vector **r** en el sistema OXYZ, conocidas en OUVW, serán:

$$\left(0, \mathbf{r}_{xyz}\right) = Q \circ \left(0, \mathbf{r}_{uvw}\right) \circ Q^* + \left(0, \mathbf{r}\right) \qquad [3.63]$$

El resultado de primero rotar y luego trasladar al sistema vendrá dado por:

$$\left(0, \mathbf{r}_{xyz}\right) = Q \circ \left(0, \mathbf{r}_{uvw} + \mathbf{p}\right) \circ Q^* \qquad [3.64]$$

Si se mantiene el sistema OXYZ fijo y se traslada el vector **r** según **p** y luego se le rota según Q se obtendrá el vector **r'** de coordenadas:

$$\left(0, \mathbf{r}'\right) = Q \circ \left(0, \mathbf{r} + \mathbf{p}\right) \circ Q^* \qquad [3.65]$$

Y si se aplica primero el giro y después la traslación **p** al vector **r**, éste se convertirá en el **r'** a través de la expresión:

$$\left(0, \mathbf{r}'\right) = Q \circ \left(0, \mathbf{r}\right) \circ Q^* + \left(0, \mathbf{p}\right) \qquad [3.66]$$

Se observa que el empleo de cuaternios para la composición de rotaciones es un método computacionalmente muy práctico, pues basta multiplicar cuaternios entre sí, lo que corresponde a una expresión de productos y sumas muy simple. Esta es su principal ventaja, tal y como se verá en la comparación de los distintos métodos.

3.5. RELACIÓN Y COMPARACIÓN ENTRE LOS DISTINTOS MÉTODOS DE LOCALIZACIÓN ESPACIAL

En los epígrafes anteriores se han explicado una serie de métodos para poder realizar la localización espacial de un sólido y de su sistema de referencia asociado. Cada uno de ellos presenta una serie de características que le hacen más o menos apto para una determinada aplicación. Así, algunos sólo sirven

para la representación de orientación, mientras otros, por ejemplo, son especialmente útiles para la composición de rotaciones.

En este apartado se analizan las ventajas e inconvenientes de cada uno de ellos, además de estudiar cómo se puede realizar el paso, cuando éste es posible, de uno a otro.

3.5.1. Comparación de métodos de localización espacial

En un principio todos los métodos expuestos son equivalentes, pero dependiendo del uso que se vaya a hacer, será más adecuado emplear un procedimiento u otro.

La comparación se realiza fundamentalmente en razón a su capacidad para la realización de cuatro cuestiones básicas de toda transformación.

1. Capacidad de representación conjunta de posición y orientación.
2. Representar la posición y orientación de un sistema rotado y trasladado O'UVW con respecto a un sistema fijo de referencia OXYZ. Que es lo mismo que representar una rotación y traslación realizada sobre un sistema de referencia.
3. Transformar un vector expresado en coordenadas con respecto a un sistema O'UVW, a su expresión en coordenadas del sistema de referencia OXYZ.
4. Rotar y trasladar un vector con respecto a un sistema de referencia fijo OXYZ.

Matrices de transformación homogénea

Sus principales ventajas residen en su capacidad de representación conjunta de posición y orientación y en la comodidad con la que se puede realizar la composición de transformaciones. Para ello bastará únicamente multiplicar, en el orden adecuado, las matrices de transformación correspondientes. Es posible, además, la aplicación de una transformación sobre un vector referido a un sistema fijo únicamente multiplicando la matriz de transformación correspondiente por el vector.

Como principal inconveniente presenta su alto nivel de redundancia (necesita definir 12 componentes para sólo 6 grados de libertad). Esto dificulta su implementación en computador.

Se trata del método más popular, pues además de trabajar con matrices, cuya álgebra es extensamente conocida, es capaz de realizar las cuatro cuestiones apuntadas anteriormente.

Ángulos de Euler

Los ángulos de Euler, en cualquiera de sus modalidades, sólo son capaces de representar orientación, y aunque permiten una notación compacta (sólo tres números reales), son difíciles de manejar para la composición de rotaciones y para su aplicación sobre un vector.

Par de rotación

El par de rotación sólo sirve para la representación de orientaciones. Es compacto, pues únicamente usa 4 parámetros para la definición de orientación de un sistema con respecto a otro. Se puede aplicar para la rotación de un vector **r** un ángulo θ alrededor del eje **k**. Sin embargo, la composición de rotaciones presenta una expresión complicada, lo que limita su utilización práctica en algunas aplicaciones.

Cuaternios

El cuaternio, como tal, sólo es capaz de representar la orientación relativa de un sistema O'UVW con respecto a otro, a través del uso de cuatro componentes. Sin embargo, como se ha visto en 3.4.2, es posible componer rotaciones junto con traslaciones de forma bastante simple y computacionalmente económica. Puede aplicarse también para la transformación de un vector tanto en traslación como en rotación.

3.5.2. Relación entre los distintos métodos de localización espacial

Ya que los métodos vistos para la representación espacial son equivalentes, es decir, expresan lo mismo de forma distinta, deberá existir un modo de pasar de un tipo de representación a otro. A continuación se muestran las relaciones de paso que se utilizan más frecuentemente. A través de ellas es posible pasar de una representación a cualquier otra, aunque en algunos casos sea más cómodo utilizar una representación auxiliar intermedia.

Ángulos de Euler: Matriz de transformación homogénea

Ya se ha mencionado en varias ocasiones que los ángulos de Euler sólo son capaces de realizar una representación de la orientación. Por ello, a la hora de obtener la matriz homogénea equivalente a un conjunto de ángulos de Euler dados, únicamente quedará definida la submatriz de rotación $\mathbf{R}_{3x3}$.

Relación directa

La obtención de la matriz homogénea correspondiente a cada conjunto de ángulos de Euler es inmediata; bastará con componer las matrices que representan las rotaciones que definen los propios ángulos:

• Sistema ZXZ

Este sistema responde, según el epígrafe 3.2.2, a la composición de la siguiente secuencia de rotaciones:

$$\mathbf{T}_{ZXZ} = \mathbf{T}(\mathbf{z},\phi)\,\mathbf{T}(\mathbf{u},\theta)\,\mathbf{T}(\mathbf{w},\psi) \qquad [3.\ 67]$$

que desarrollado en forma matricial:

$$\mathbf{T} = \mathbf{T}(\mathbf{z},\phi)\,\mathbf{T}(\mathbf{u},\theta)\,\mathbf{T}(\mathbf{w},\psi) = \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\theta & -S\theta & 0 \\ 0 & S\theta & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 & 0 \\ S\psi & C\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} C\phi C\psi - S\phi C\theta S\psi & -C\phi S\psi - S\phi C\theta C\psi & S\phi S\theta & 0 \\ S\phi C\psi + C\phi C\theta S\psi & -S\phi S\psi + C\phi C\theta C\psi & -C\phi S\theta & 0 \\ S\theta S\psi & S\theta C\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 68]$$

- Sistema ZYZ

Según el epígrafe 3.2.2, el paso del sistema fijo al girado se hace realizando la siguiente secuencia de rotaciones:

$$\mathbf{T} = \mathbf{T}(\mathbf{z},\phi)\,\mathbf{T}(\mathbf{v},\theta)\,\mathbf{T}(\mathbf{w},\psi) \qquad [3.\ 69]$$

que desarrollado en forma matricial:

$$\mathbf{T} = \mathbf{T}(z,\phi)\,\mathbf{T}(v,\theta)\,\mathrm{T}(w,\psi) = \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 & 0 \\ S\psi & C\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} C\phi C\theta C\psi - S\phi S\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta & 0 \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta & 0 \\ -S\theta C\psi & S\theta S\psi & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 70]$$

- Roll-Pitch-Yaw

De igual forma que en los casos anteriores, estos ángulos de Euler se pueden representar mediante la concatenación de las siguientes rotaciones:

$$\mathbf{T} = \mathbf{T}(\mathbf{z},\phi)\,\mathbf{T}(\mathbf{y},\theta)\,\mathbf{T}(\mathbf{x},\psi) \qquad [3.\ 71]$$

y de forma matricial:

$$\mathbf{T} = \mathbf{T}(\mathbf{z},\phi)\,\mathbf{T}(\mathbf{y},\theta)\,\mathbf{T}(\mathbf{x},\psi) = \begin{bmatrix} C\phi & -S\phi & 0 & 0 \\ S\phi & C\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta & 0 \\ 0 & 1 & 0 & 0 \\ -S\theta & 0 & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\psi & -S\psi & 0 \\ 0 & S\psi & C\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} C\phi C\theta & C\phi S\theta S\psi - S\phi C\psi & C\phi S\theta C\psi + S\phi S\psi & 0 \\ S\phi C\theta & S\phi S\theta S\psi + C\phi C\psi & S\phi S\theta C\psi - C\phi S\psi & 0 \\ -S\theta & C\theta S\psi & C\theta C\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.\ 72]$$

Relación inversa

El paso de la representación mediante matriz homogénea a cualquiera de los conjuntos de ángulos de Euler vistos no es trivial, pues se necesita resolver una serie de ecuaciones trigonométricas acopladas.

Par de rotación: Matriz de transformación homogénea

Al igual que en el caso de los ángulos de Euler, mediante un eje y ángulo de rotación sólo es posible representar orientación; de ahí que únicamente quede definida la submatriz de rotación $\mathbf{R}_{3x3}$ de la matriz homogénea de transformación.

Relación directa

Se quiere descomponer el giro de un ángulo θ alrededor del eje $\mathbf{k}(k_x,k_y,k_z)$ en la composición de rotaciones básicas que se puedan expresar mediante matrices básicas de rotación. Esto se consigue realizando una serie de rotaciones para alinear el eje **k** con uno de los ejes coordenados, por ejemplo el OZ, girar el ángulo θ con respecto a él y deshacer las rotaciones previas hasta tener el vector **k** en su posición inicial. Viendo la Figura 3.20, esto se consigue mediante la siguiente composición:

$$\mathbf{T}(\mathbf{k},\theta) = \mathbf{T}(\mathbf{x},-\alpha)\mathbf{T}(\mathbf{y},\beta)\mathbf{T}(\mathbf{z},\theta)\mathbf{T}(\mathbf{y},-\beta)\mathbf{T}(\mathbf{x},\alpha) \qquad [3.73]$$

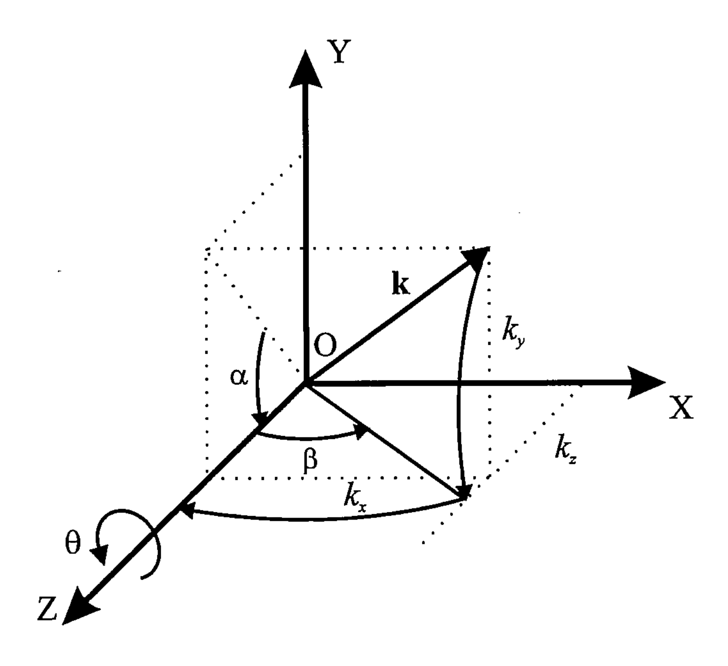


Figura 3.20. Ángulo y eje de rotación.

Teniendo en cuenta las siguientes relaciones:

$$\text{sen}\alpha = \frac{k_y}{\sqrt{k_y^2 + k_z^2}} \qquad \cos\alpha = \frac{k_z}{\sqrt{k_y^2 + k_z^2}}$$
$$\text{sen}\beta = k_x \qquad \cos\beta = \sqrt{k_y^2 + k_z^2} \tag{3.74}$$

Expresándolo en forma matricial y sustituyendo en la ecuación [3.73] se tiene la matriz homogénea que expresa la rotación alrededor del eje **k** de un ángulo θ:

$$\mathbf{T}(\mathbf{k},\theta) = \begin{bmatrix} k_x^2\,V\theta + C\theta & k_x k_y\,V\theta - k_z S\theta & k_x k_z\,V\theta + k_y\,S\theta & 0 \\ k_x k_y\,V\theta + k_z\,S\theta & k_y^2\,V\theta + C\theta & k_y k_z\,V\theta - k_x\,S\theta & 0 \\ k_x k_z\,V\theta - k_y\,S\theta & k_y k_z\,V\theta + k_x S\theta & k_z^2\,V\theta + C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.75}$$

donde V representa el término *verseno* definido como Vθ = 1 - Cθ.

<u>Relación inversa</u>

Se quiere obtener un eje **k** y un ángulo θ de rotación equivalente a la representación de una rotación mediante la matriz homogénea de rotación:

$$\mathbf{Rot} = \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.76}$$

Se podrán igualar las matrices de las expresiones [3.75] [3.76] y realizar una equivalencia componente a componente. Considerando el ángulo θ como positivo entre 0° y 180°, se puede llegar a las siguientes expresiones tanto para el ángulo θ como para las componentes del vector:

$$\text{tg}\theta = \frac{\sqrt{(o_z - a_y)^2 + (a_x - n_z)^2 + (n_y - o_x)^2}}{(n_x + o_y + a_z - 1)}$$
$$k_x = \frac{o_z - a_y}{2\text{sen}\theta} \qquad k_y = \frac{a_x - n_z}{2\text{sen}\theta} \qquad k_z = \frac{n_y - o_x}{2\text{sen}\theta} \tag{3.77}$$

Cuando el ángulo θ se acerca a 0° o a 180° aparecen problemas de indeterminación en las ecuaciones anteriores. Por ello, en estos casos es necesario realizar otro tipo de aproximación. Ésta se describe con detalle en [PAUL-81].

Par de rotación: Cuaternios

Relación directa

Por la propia definición de cuaternios dada en el epígrafe 3.2.4, un cuaternio Q se puede expresar como:

$$Q = \left(cos\frac{\theta}{2}, \mathbf{k}\,\text{sen}\frac{\theta}{2} \right) \quad [3.\ 78]$$

y componente a componente:

$$\begin{array}{ll} q_0 = cos\frac{\theta}{2} & q_1 = k_x\,\text{sen}\frac{\theta}{2} \\ q_2 = k_y\,\text{sen}\frac{\theta}{2} & q_3 = k_z\,\text{sen}\frac{\theta}{2} \end{array} \quad [3.\ 79]$$

Relación inversa

La relación inversa se obtiene fácilmente de las anteriores expresiones:

$$\begin{array}{l} \theta = 2\,\text{arcos}(q_0) \\ k_x = \dfrac{q_1}{(1-q_0^2)^{1/2}} \quad k_y = \dfrac{q_2}{(1-q_0^2)^{1/2}} \quad k_z = \dfrac{q_3}{(1-q_0^2)^{1/2}} \end{array} \quad [3.\ 80]$$

Cuaternios: Matriz de transformación homogénea

El paso de cuaternios a la matriz de transformación homogénea, y viceversa, se pueden deducir fácilmente utilizando como representación auxiliar intermedia el eje y ángulo de rotación. A continuación se expresan las relaciones finales, obviando los desarrollos intermedios que se pueden encontrar en [HAMILTON-69].

Relación directa

La representación de la matriz de transformación **T** en función de las componentes de un cuaternio Q viene dada por la siguiente matriz [BEGGS-83]:

$$\mathbf{T} = 2\begin{bmatrix} q_0^2+q_1^2-\frac{1}{2} & q_1q_2-q_3q_0 & q_1q_3+q_2q_0 & 0 \\ q_1q_2+q_3q_0 & q_0^2+q_2^2-\frac{1}{2} & q_2q_3-q_1q_0 & 0 \\ q_1q_3-q_2q_0 & q_2q_3+q_1q_0 & q_0^2+q_3^2-\frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [3.81]$$

Relación inversa

La relación inversa se puede obtener fácilmente igualando la traza y los elementos de la diagonal principal de la matriz [3.81] con la de la matriz [3.76]:

$$\begin{aligned} q_0 &= \frac{1}{2}\sqrt{\left(n_x+o_y+a_z+1\right)} \\ q_1 &= \frac{1}{2}\sqrt{\left(n_x-o_y-a_z+1\right)} \\ q_2 &= \frac{1}{2}\sqrt{\left(-n_x+o_y-a_z+1\right)} \\ q_3 &= \frac{1}{2}\sqrt{\left(-n_x-o_y+a_z+1\right)} \end{aligned} \qquad [3.82]$$

Tabla 3.2. Relación entre los métodos de localización espacial

	Matriz homogénea	Cuaternios	Par de rotación
Matriz homogénea	-	[3.82]	[3.77]
Cuaternios	[3.81]	-	[3.80]
Par de rotación	[3.75]	[3.79]	-
Euler ZXZ	[3.68]	-	-
Euler ZYZ	[3.70]	-	-
Euler RPY	[3.72]	-	-

La Tabla 3.2 recoge de forma condensada las relaciones posibles entre los distintos métodos de localización espacial y cuáles son las expresiones que los relacionan.

3.6. BIBLIOGRAFÍA

[BARRIENTOS-86] A. Barrientos, "Software de base para un robot de la segunda generación", en: *Sistema operativo y lenguaje de programación para robots dotados de sensores*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1986.

[BEGGS-83] J.S. Beggs, *Kinematics*, Springer Verlag, Berlin, 1983.

[CORKE-96] P.I Corke, "A Robotics Toolbox for Matlab", *IEEE Robotics & Automation Magazine*, Vol. 3, no. 1, marzo 1996.

[FU-88] Fu, K.S., González, R.C. y Lee, C.S.G., "Cinemática del brazo del robot", en: *Robótica: control, detección, visión e inteligencia*, McGraw-Hill, Madrid, 1988.

[HAMILTON-69] Hamilton, W.R., *Elements of Quaternions*, Chelsea Publishing Co. New York, 1969.

[PAUL-81] Paul, R.P., "Homogeneous Transformations", en: *Robot Manipulators: Mathematics, Programming and Control*, The MIT Press, Massachussets, 1981.

Capítulo 4

CINEMÁTICA DEL ROBOT

Quien bien ata, bien desata.
Refranero popular

La cinemática del robot estudia el movimiento del mismo con respecto a un sistema de referencia. Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo, y en particular por las relaciones entre la posición y la orientación del extremo final del robot con los valores que toman sus coordenadas articulares.

Existen dos problemas fundamentales a resolver en la cinemática del robot (Figura 4.1); el primero de ellos se conoce como el **problema cinemático directo**, y consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto a un sistema de coordenadas que se toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos de los elementos del robot; el segundo, denominado **problema cinemático inverso**, resuelve la configuración que debe adoptar el robot para una posición y orientación del extremo conocidas.

Denavit y Hartenberg propusieron un método sistemático para describir y representar la geometría espacial de los elementos de una cadena cinemática, y en particular de un robot, con respecto a un sistema de referencia fijo. Este método utiliza una matriz de transformación homogénea para describir la relación espacial entre dos elementos rígidos adyacentes, reduciéndose el problema cinemático directo a encontrar una matriz de transformación homogénea 4 x 4 que relacione la localización espacial del extremo del robot con respecto al sistema de coordenadas de su base.

Por otra parte, la cinemática del robot trata también de encontrar las relaciónes entre las velocidades del movimiento de las articulaciones y las del extremo. Esta relación viene dada por el **modelo diferencial** expresado mediante la matriz Jacobiana.

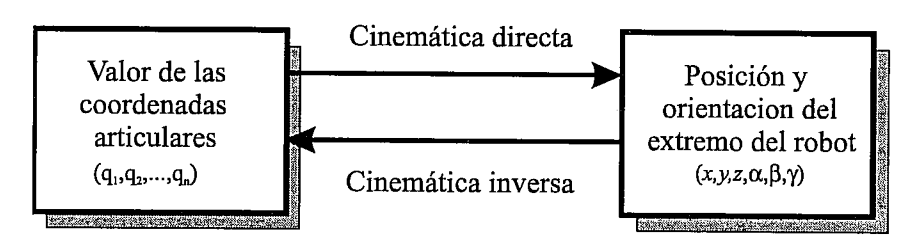


Figura 4.1. Diagrama de relación entre cinemática directa e inversa.

4.1. EL PROBLEMA CINEMÁTICO DIRECTO

Como se ha visto en el capítulo anterior, se utiliza fundamentalmente el álgebra vectorial y matricial para representar y describir la localización de un objeto en el espacio tridimensional con respecto a un sistema de referencia fijo. Dado que un robot se puede considerar como una cadena cinemática formada por objetos rígidos o *eslabones* unidos entre sí mediante *articulaciones*, se puede establecer un sistema de referencia fijo situado en la base del robot y describir la localización de cada uno de los eslabones con respecto a dicho sistema de referencia. De esta forma, el problema cinemático directo se reduce a encontrar una matriz homogénea de transformación **T** que relacione la posición y orientación del extremo del robot respecto del sistema de referencia fijo situado en la base del mismo. Esta matriz **T** será función de las coordenadas articulares.

4.1.1. Resolución del problema cinemático directo mediante matrices de transformación homogénea

La resolución del problema cinemático directo consiste en encontrar las relaciones que permiten conocer la localización espacial del extremo del robot a partir de los valores de sus coordenadas articulares.

Así, si se han escogido coordenadas cartesianas y ángulos de Euler para representar la posición y orientación del extremo de un robot de seis grados de libertad, la solución al problema cinemático directo vendrá dada por las relaciones:

$$
\begin{aligned}
x &= \mathrm{f}_x(q_1,q_2,q_3,q_4,q_5,q_6)\\
y &= \mathrm{f}_y(q_1,q_2,q_3,q_4,q_5,q_6)\\
z &= \mathrm{f}_z(q_1,q_2,q_3,q_4,q_5,q_6)\\
\alpha &= \mathrm{f}_\alpha(q_1,q_2,q_3,q_4,q_5,q_6)\\
\beta &= \mathrm{f}_\beta(q_1,q_2,q_3,q_4,q_5,q_6)\\
\gamma &= \mathrm{f}_\gamma(q_1,q_2,q_3,q_4,q_5,q_6)
\end{aligned}
\qquad [4.1]
$$

La obtención de estas relaciones no es en general complicada, siendo incluso en ciertos casos (robots de pocos GDL) fácil de encontrar mediante simples consideraciones geométricas. Por ejemplo, para el caso de un robot con 2 GDL (Figura 4.2), es fácil comprobar que:

$$\begin{aligned} x &= l_1 \cos q_1 + l_2 \cos(q_1 + q_2) \\ y &= l_1 \operatorname{sen} q_1 + l_2 \operatorname{sen}(q_1 + q_2) \end{aligned} \qquad [4.2]$$

Para robots de más grados de libertad puede plantearse un método sistemático basado en la utilización de las matrices de transformación homogénea.

En general, un robot de n grados de libertad está formado por n eslabones unidos por n articulaciones, de forma que cada par articulación-eslabón constituye un grado de libertad. A cada eslabón se le puede asociar un sistema de referencia solidario a él y, utilizando las transformaciones homogéneas, es posible representar las rotaciones y traslaciones relativas entre los distintos eslabones que componen el robot. Normalmente, la matriz de transformación homogénea que representa la posición y orientación relativa entre los sistemas asociados a dos eslabones consecutivos del robot se suele denominar matriz ${}^{i-1}\mathbf{A}_i$. Así pues, ${}^{0}\mathbf{A}_1$ describe la posición y orientación del sistema de referencia solidario al primer eslabón con respecto al sistema de referencia solidario a la base, ${}^{1}\mathbf{A}_2$ describe la posición y orientación del segundo eslabón respecto del primero, etc. Del mismo modo, denominando ${}^{0}\mathbf{A}_k$ a las matrices resultantes del producto de las matrices ${}^{i-1}\mathbf{A}_i$ con i desde 1 hasta k, se puede representar de forma total o parcial la cadena cinemática que forma el robot. Así, por ejemplo, la posición y orientación del sistema solidario con el segundo eslabón del robot con respecto al sistema de coordenadas de la base se puede expresar mediante la matriz ${}^{0}\mathbf{A}_2$:

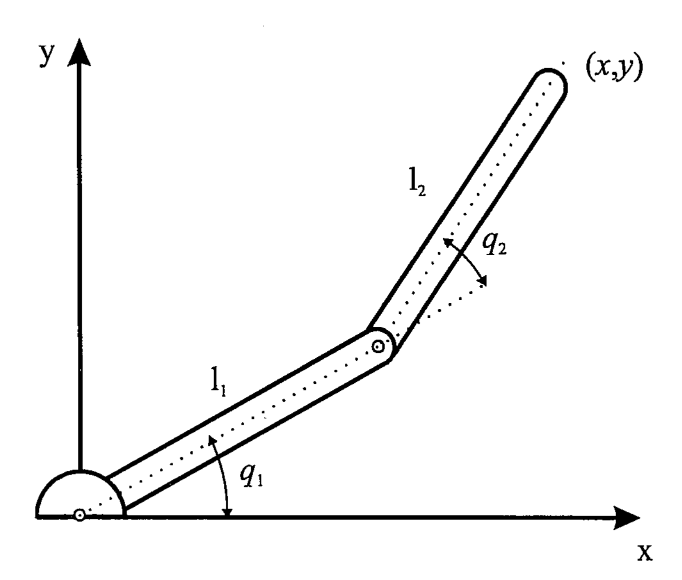


Figura 4.2. Robot planar de 2 grados de libertad.

$$ {}^{0}\mathbf{A}_2 = {}^{0}\mathbf{A}_1\, {}^{1}\mathbf{A}_2 \qquad [4.\ 3] $$

De manera análoga, la matriz ${}^{0}\mathbf{A}_3$ representa la localización del sistema del tercer eslabón:

$$ {}^{0}\mathbf{A}_3 = {}^{0}\mathbf{A}_1\, {}^{1}\mathbf{A}_2\, {}^{2}\mathbf{A}_3 \qquad [4.\ 4] $$

Cuando se consideran todos los grados de libertad, a la matriz ${}^{0}\mathbf{A}_n$ se le suele denominar **T**. Así, dado un robot de seis grados de libertad, se tiene que la posición y orientación del eslabón final vendrá dada por la matriz **T** :

$$ \mathbf{T} = {}^{0}\mathbf{A}_6 = {}^{0}\mathbf{A}_1\, {}^{1}\mathbf{A}_2\, {}^{2}\mathbf{A}_3\, {}^{3}\mathbf{A}_4\, {}^{4}\mathbf{A}_5\, {}^{5}\mathbf{A}_6 \qquad [4.\ 5] $$

Aunque para describir la relación que existe entre dos elementos contiguos se puede hacer uso de cualquier sistema de referencia ligado a cada elemento, la forma habitual que se suele utilizar en robótica es la representación de Denavit-Hartenberg (D-H). Denavit y Hartenberg [DENAVIT-55] propusieron en 1955 un método matricial que permite establecer de manera sistemática un sistema de coordenadas $\{S_i\}$ ligado a cada eslabón i de una cadena articulada, pudiéndose determinar a continuación las ecuaciones cinemáticas de la cadena completa [PAUL-81].

Según la representación de D-H, escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, será posible pasar de uno al siguiente mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón.

Estas transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permiten relacionar el sistema de referencia del elemento i con el sistema del elemento i-1. Las transformaciones en cuestión son las siguientes (es importante recordar que el paso del sistema $\{S_{i-1}\}$ al $\{S_i\}$ mediante estas 4 transformaciones está garantizado sólo si los sistemas $\{S_{i-1}\}$y $\{S_i\}$ han sido definidos de acuerdo a unas normas determinadas que se expondrán posteriormente):

1. Rotación alrededor del eje $\mathbf{z}_{i-1}$ un ángulo θ_i.
2. Traslación a lo largo de $\mathbf{z}_{i-1}$ una distancia d_i ; vector $\mathbf{d}_i\,(0,0,d_i)$.
3. Traslación a lo largo de $\mathbf{x}_i$ una distancia a_i ; vector $\mathbf{a}_i\,(0,0,a_i)$.
4. Rotación alrededor del eje $\mathbf{x}_i$ un ángulo α_i.

Dado que el producto de matrices no es conmutativo, las transformaciones se han de realizar en el orden indicado. De este modo se tiene que:

$$ {}^{i-1}\mathbf{A}_i = \mathbf{T}(\mathbf{z},\theta_i)\ \mathbf{T}(0,0,d_i)\ \mathbf{T}(a_i,0,0)\ \mathbf{T}(\mathbf{x},\alpha_i) \qquad [4.\ 6] $$

y realizando el producto entre matrices:

$$
{}^{i-1}\mathbf{A}_i = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =
$$

$$
= \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [4.7]
$$

donde θ_i , a_i , d_i , α_i son los parámetros D-H del eslabón i. De este modo, basta con identificar los parámetros θ_i , a_i , d_i , α_i para obtener las matrices **A** y relacionar así todos y cada uno los eslabones del robot.

Como se ha indicado, para que la matriz ${}^{i-1}\mathbf{A}_i$, definida en [4.7] relacione los sistemas $\{S_i\}$ y $\{S_{i-1}\}$, es necesario que los sistemas se hayan escogido de acuerdo a unas determinadas normas. Éstas, junto con la definición de los 4 parámetros de Denavit Hartenberg, conforman el siguiente algoritmo para la resolución del problema cinemático directo:

4.1.2. Algoritmo de Denavit-Hartenberg para la obtención del modelo cinemático directo

D-H 1. Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot.

D-H 2. Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad) y acabando en n.

D-H 3. Localizar el eje de cada articulación. Si ésta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.

D-H 4. Para i de 0 a n-1 situar el eje $\mathbf{z}_i$ sobre el eje de la articulación i+1.

D-H 5. Situar el origen del sistema de la base $\{S_0\}$ en cualquier punto del eje $\mathbf{z}_0$. Los ejes $\mathbf{x}_0$ e $\mathbf{y}_0$ se situarán de modo que formen un sistema dextrógiro con $\mathbf{z}_0$.

D-H 6. Para i de 1 a n-1, situar el sistema $\{S_i\}$ (solidario al eslabón i) en la intersección del eje $\mathbf{z}_i$ con la línea normal común a $\mathbf{z}_{i-1}$ y $\mathbf{z}_i$. Si ambos ejes se cortasen se situaría $\{S_i\}$ en el punto de corte. Si fuesen paralelos $\{S_i\}$ se situaría en la articulación i+1.

D-H 7. Situar $\mathbf{x}_i$ en la línea normal común a $\mathbf{z}_{i-1}$ y $\mathbf{z}_i$.

D-H 8. Situar $\mathbf{y}_i$ de modo que forme un sistema dextrógiro con $\mathbf{x}_i$ y $\mathbf{z}_i$.

D-H 9. Situar el sistema $\{S_n\}$ en el extremo del robot de modo que $\mathbf{z}_n$ coincida con la dirección de $\mathbf{z}_{n-1}$ y $\mathbf{x}_n$ sea normal a $\mathbf{z}_{n-1}$ y $\mathbf{z}_n$.

D-H 10. Obtener θ_i como el ángulo que hay que girar en torno a $\mathbf{z}_{i-1}$ para que $\mathbf{x}_{i-1}$ y $\mathbf{x}_i$ queden paralelos.

D-H 11. Obtener d_i como la distancia, medida a lo largo de $\mathbf{z}_{i-1}$, que habría que desplazar $\{S_{i-1}\}$ para que $\mathbf{x}_i$ y $\mathbf{x}_{i-1}$ quedasen alineados.

DH 12. Obtener a_i como la distancia medida a lo largo de $\mathbf{x}_i$ (que ahora coincidiría con $\mathbf{x}_{i-1}$) que habría que desplazar el nuevo $\{S_{i-1}\}$ para que su origen coincidiese con $\{S_i\}$.

DH 13. Obtener α_i como el ángulo que habría que girar entorno a $\mathbf{x}_i$ (que ahora coincidiría con $\mathbf{x}_{i-1}$), para que el nuevo $\{S_{i-1}\}$ coincidiese totalmente con $\{S_i\}$.

DH 14. Obtener las matrices de transformación ${}^{i-1}\mathbf{A}_i$ definidas en [4.7].

DH 15. Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot $\mathbf{T} = {}^{0}\mathbf{A}_1, {}^{1}\mathbf{A}_2 .. {}^{n-1}\mathbf{A}_n$.

DH 16. La matriz **T** define la orientación (submatriz de rotación) y posición (submatriz de traslación) del extremo referido a la base en función de las n coordenadas articulares.

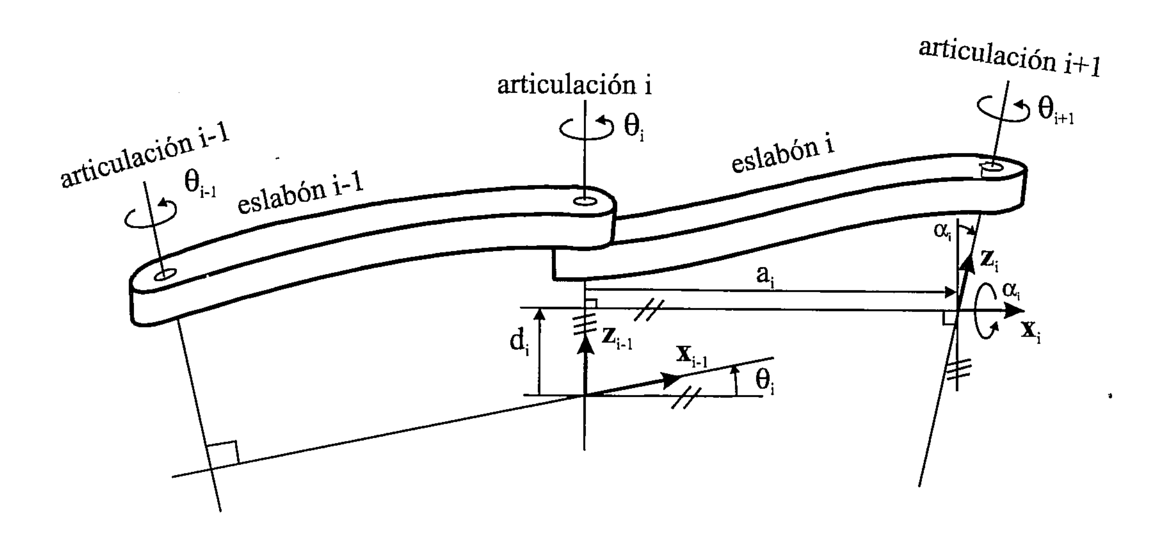


Figura 4.3. Parámetros D-H para un eslabón giratorio.

Los cuatro parámetros de D-H (θ_i, d_i, a_i, α_i) dependen únicamente de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente. En concreto estos representan (Figura 4.3):

- θ_i Es el ángulo que forman los ejes $\mathbf{x}_{i-1}$ y $\mathbf{x}_i$ medido en un plano perpendicular al eje $\mathbf{z}_{i-1}$, utilizando la regla de la mano derecha. Se trata de un parámetro variable en articulaciones giratorias.
- d_i Es la distancia a lo largo del eje $\mathbf{z}_{i-1}$ desde el origen del sistema de coordenadas (i-1)-ésimo hasta la intersección del eje $\mathbf{z}_{i-1}$ con el eje $\mathbf{x}_i$. Se trata de un parámetro variable en articulaciones prismáticas.
- a_i Es la distancia a lo largo del eje $\mathbf{x}_i$ que va desde la intersección del eje $\mathbf{z}_{i-1}$ con el eje $\mathbf{x}_i$ hasta el origen del sistema i-ésimo, en el caso de articulaciones giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes $\mathbf{z}_{i-1}$ y $\mathbf{z}_i$.
- α_i Es el ángulo de separación del eje $\mathbf{z}_{i-1}$ y el eje $\mathbf{z}_i$, medido en un plano perpendicular al eje $\mathbf{x}_i$, utilizando la regla de la mano derecha.

Una vez obtenidos los parámetros D-H, el cálculo de las relaciones entre los eslabones consecutivos del robot es inmediato, ya que vienen dadas por las matrices **A**, que se calculan según la expresión general [4.7]. Las relaciones entre eslabones no consecutivos vienen dadas por las matrices **T** que, como ya se comentó anteriormente, se obtienen como producto de un conjunto de matrices **A**.

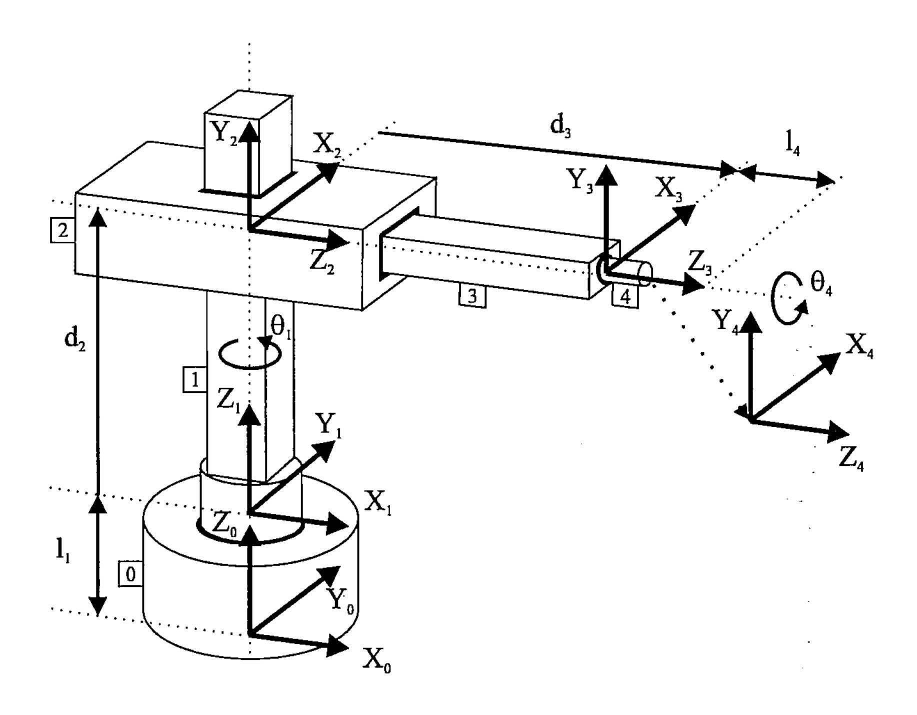


Figura 4.4. Robot cilíndrico del Ejemplo 4.1.

Obtenida la matriz **T**, ésta expresará la orientación (submatriz (3 x 3) de rotación) y posición (submatriz (3 x 1) de traslación) del extremo del robot en función de sus coordenadas articulares, con lo que quedará resuelto el problema cinemático directo.

Tabla 4.1. Parámetros D-H para el robot cilíndrico de la Figura 4.4.

Articulación	θ	d	a	α
1	q_1	l_1	0	0
2	90	d_2	0	90
3	0	d_3	0	0
4	q_4	l_4	0	0

Ejemplo 4.1.

Con el fin de ilustrar el método expuesto anteriormente, se va a desarrollar a continuación la resolución completa del problema cinemático directo para un robot cilíndrico.

En primer lugar, se localizan los sistemas de referencia de cada una de las articulaciones del robot (Figura 4.4). Posteriormente se determinan los parámetros de Denavit-Hartenberg del robot, con los que se construye la Tabla 4.1. Una vez calculados los parámetros de cada eslabón, se calculan las matrices **A**, sustituyendo en la expresión general [4.7] de la siguiente manera :

$$^{0}\mathbf{A}_1 = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad ^{1}\mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{2}\mathbf{A}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad ^{3}\mathbf{A}_4 = \begin{bmatrix} C_4 & -S_4 & 0 & 0 \\ S_4 & C_4 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Así pues, se puede calcular la matriz **T** que indica la localización del sistema final con respecto al sistema de referencia de la base del robot.

$$\mathbf{T} = {}^{0}\mathbf{A}_1\, {}^{1}\mathbf{A}_2\, {}^{2}\mathbf{A}_3\, {}^{3}\mathbf{A}_4 = \begin{bmatrix} -S_1C_4 & S_1S_4 & C_1 & C_1(d_3+l_4) \\ C_1C_4 & -C_1S_4 & S_1 & S_1(d_3+l_4) \\ S_4 & C_4 & 0 & d_2+l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Tabla 4.2. Parámetros D-H para un robot IRB6400C

Articulación	θ	d	a	α
1	θ_1	0	0	-90
2	θ_2	l_1	0	90
3	θ_3-90	0	$-l_2$	90
4	θ_4	l_3	0	-90
5	θ_5	0	0	90
6	θ_6	l_4	0	0

Ejemplo 4.2.

Se va a desarrollar a continuación la resolución completa del problema cinemático directo para un robot IRB6400C.

En primer lugar, y siguiendo el algoritmo de Denavit-Hatenberg, se localizan los sistemas de referencia de cada una de las articulaciones del robot (Figura 4.5).Posteriormente se determinan los parámetros de Denavit-Hartenberg del robot, con los que se construye la Tabla 4.2. Se calculan ahora las matrices **A**, sustituyendo en la expresión general de la siguiente manera :

$$^{0}\mathbf{A}_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad ^{1}\mathbf{A}_2 = \begin{bmatrix} C_2 & 0 & S_2 & 0 \\ S_2 & 0 & -C_2 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{2}\mathbf{A}_3 = \begin{bmatrix} S_3 & 0 & -C_3 & -l_2S_3 \\ -C_3 & 0 & -S_3 & l_2C_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad ^{3}\mathbf{A}_4 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^{4}\mathbf{A}_5 = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad ^{5}\mathbf{A}_6 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & l_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

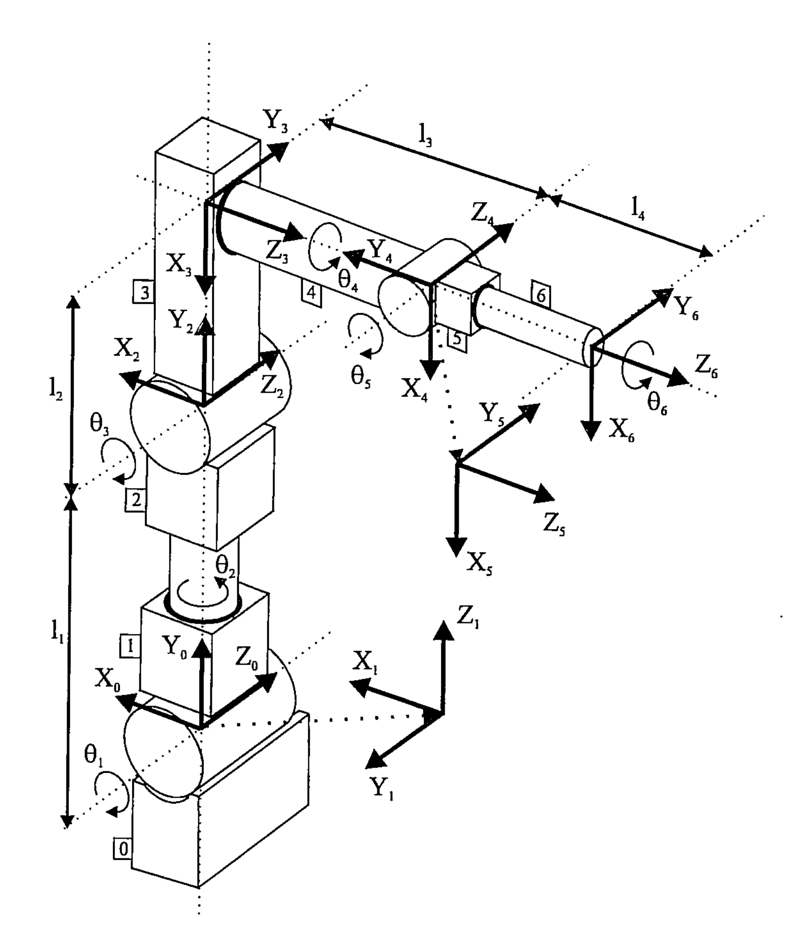


Figura 4.5. Asignación de ejes de referencia para el robot IRB6400C del Ejemplo 4.2.

Así pues, se puede calcular la matriz **T** que indica la localización del sistema asociado al extremo del robot con respecto al sistema de referencia de la base del robot:

$$\mathbf{T} = {}^0\mathbf{A}_1\,{}^1\mathbf{A}_2\,{}^2\mathbf{A}_3\,{}^3\mathbf{A}_4\,{}^4\mathbf{A}_5\,{}^5\mathbf{A}_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A continuación se desarrollan los términos de la matriz **T**.

$$
\begin{aligned}
n_x &= (C_1C_2S_3+S_1C_3)(C_4C_5C_6-S_4S_6)+C_1S_2(S_4C_5C_6+C_4S_6)+(-C_1C_2C_3+S_1S_3)S_5C_6\\
n_y &= (-S_1C_2S_3+S_1C_3)(C_4C_5C_6-S_4S_6)+S_1S_2(S_4C_5C_6+C_4S_6)+(-S_1C_2C_3-C_1S_3)S_5C_6\\
n_z &= (-S_2S_3)(C_4C_5C_6-S_4S_6)+C_2(S_4C_5C_6+C_4S_6)+S_2C_3S_5C_6\\
o_x &= (C_1C_2S_3+S_1C_3)(-C_4C_5C_6-S_4S_6)+C_1S_2(-S_4C_5C_6+C_4S_6)+(-C_1C_2C_3+S_1S_3)(-S_5C_6)\\
o_y &= (-S_1C_2S_3+S_1C_3)(-C_4C_5C_6-S_4S_6)+S_1S_2(-S_4C_5C_6+C_4S_6)+(-S_1C_2C_3-C_1S_3)(-S_5C_6)\\
o_z &= (-S_2S_3)(-C_4C_5C_6-S_4S_6)+C_2(-S_4C_5C_6+C_4S_6)+S_2C_3(-S_5C_6)\\
p_x &= (C_1C_2S_3+S_1C_3)(l_4C_4S_5)+C_1S_2(l_4S_4S_5)+(C_1C_2C_3+S_1S_3)(-l_4C_5+l_3)+\\
&\qquad (-l_2C_1C_2S_3-l_2S_1C_3-l_1S_1)\\
p_y &= (-S_1C_2S_3-C_1C_3)(l_4C_4S_5)+S_1S_2(l_4S_4S_5)+(-C_1C_2C_3-C_1S_3)(-l_4C_5+l_3)+\\
&\qquad (-l_2S_1C_2S_3-l_2C_1C_3+l_1C_1)\\
p_z &= (-S_2S_3)(l_4C_4C_5)+C_2(l_4S_4S_5)+S_2C_3(-l_4C_5+l_3)+l_2S_2S_3
\end{aligned}
$$

Para calcular las ecuaciones anteriores se necesita realizar 12 llamadas a funciones transcendentes, además de numerosas sumas y productos, teniendo en cuenta que el vector **a** se calcula como el producto vectorial de los vectores **n** y **o** (**a** = **n** x **o**).

Como se ve, en estas ecuaciones queda reflejado el valor de la posición (p_x, p_y, p_z) y orientación (**n**,**o**,**a**) del extremo del robot en función de las coordenadas articulares ($\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6$).

4.1.3. Resolución del problema cinemático directo mediante el uso de cuaternios

Puesto que las matrices de transformación homogénea y los cuaternios son métodos alternativos para representar transformaciones de rotación y desplazamiento, será posible utilizar estos últimos de manera equivalente a las matrices para la resolución del problema cinemático directo de un robot.

Para aclarar el uso de los cuaternios con este fin, se van a utilizar a continuación para resolver el problema cinemático directo de un robot tipo SCARA cuya estructura geométrica se representa en la Figura 4.6.

El procedimiento a seguir será el de obtener la expresión que permite conocer las coordenadas de la posición y orientación del sistema de referencia asociado al extremo del robot $\{S_4\}$ con respecto al sistema de referencia asociado a la base $\{S_0\}$. Esta relación será función de las longitudes l_1, l_2 y l_3 de los elementos del robot así como de las coordenadas articulares q_1, q_2, q_3 y q_4.

Para obtener la relación entre $\{S_0\}$ y $\{S_4\}$ se irá convirtiendo sucesivamente $\{S_0\}$ en $\{S_1\}$, $\{S_2\}$, $\{S_3\}$ y $\{S_4\}$ según la siguiente serie de transformaciones:

1. Desplazamiento de $\{S_0\}$ una distancia l_1 a lo largo del eje $\mathbf{z}_0$ y giro un ángulo q_1 alrededor del eje $\mathbf{z}_0$, llegándose a $\{S_1\}$.
2. Desplazamiento de $\{S_1\}$ una distancia l_2 a lo largo del eje $\mathbf{x}_1$ y giro un ángulo q_2 alrededor del nuevo eje $\mathbf{z}$, para llegar al sistema $\{S_2\}$.
3. Desplazamiento a lo largo del eje $\mathbf{x}_2$ una distancia l_3 para llegar al sistema $\{S_3\}$.
4. Desplazamiento de $\{S_3\}$ una distancia q_3 a lo largo del eje $\mathbf{z}_3$ y giro en torno a $\mathbf{z}_4$ de un ángulo q_4, llegándose finalmente a $\{S_4\}$.

De manera abreviada las sucesivas transformaciones quedan representadas por:

$$\begin{array}{lll} S_0 \to S_1: & \mathbf{T}(\mathbf{z},l_1) & \mathbf{Rot}(\mathbf{z},q_1) \\ S_1 \to S_2: & \mathbf{T}(\mathbf{x},l_2) & \mathbf{Rot}(\mathbf{z},q_2) \\ S_2 \to S_3: & \mathbf{T}(\mathbf{x},l_3) & \mathbf{Rot}(\mathbf{z},0) \\ S_3 \to S_4: & \mathbf{T}(\mathbf{z},-q_3) & \mathbf{Rot}(\mathbf{z},q_4) \end{array} \quad [4.8]$$

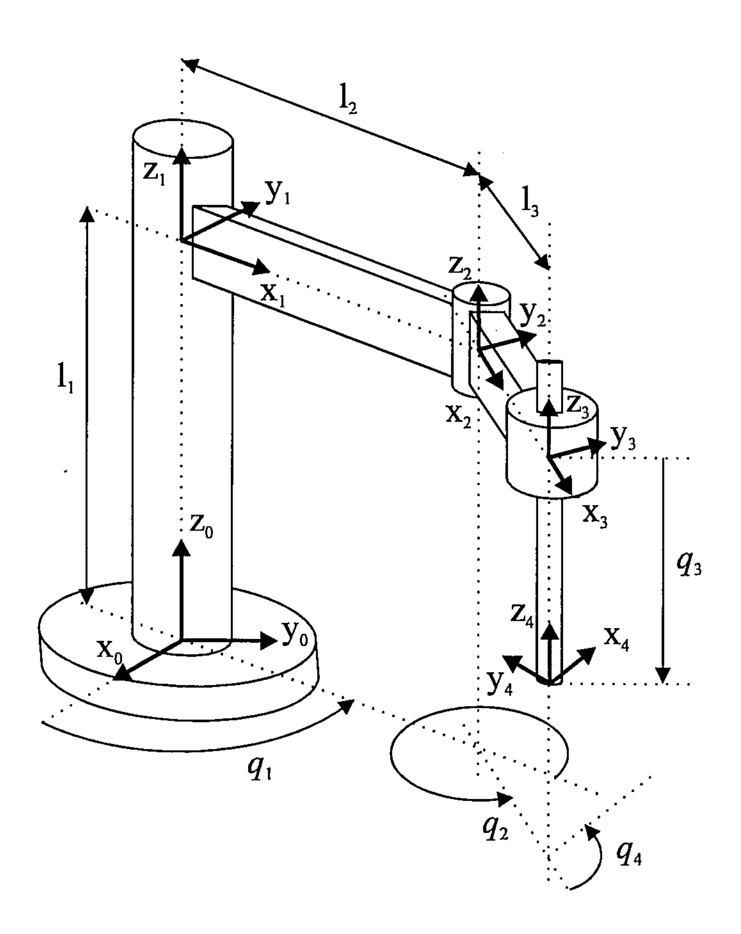


Figura 4.6. Asignación de sistemas de referencia en un robot SCARA.

donde los desplazamientos quedan definidos por los vectores:

$$\begin{aligned} \mathbf{p}_1 &= (0,0,l_1) \\ \mathbf{p}_2 &= (l_2,0,0) \\ \mathbf{p}_3 &= (l_3,0,0) \\ \mathbf{p}_4 &= (0,0,-q_3) \end{aligned} \qquad [4.\ 9]$$

y los giros por los cuaternios:

$$\begin{aligned} Q_1 &= (\hat{C}_1,0,0,\hat{S}_1) \\ Q_2 &= (\hat{C}_2,0,0,\hat{S}_2) \\ Q_3 &= (1,0,0,0) \\ Q_4 &= (\hat{C}_4,0,0,\hat{S}_4) \end{aligned} \qquad [4.\ 10]$$

donde:

$$\begin{aligned} \hat{C}_i &= \cos\left(\frac{q_i}{2}\right) \\ \hat{S}_i &= \mathrm{sen}\left(\frac{q_i}{2}\right) \end{aligned} \qquad [4.11]$$

Aplicando las ecuaciones [3.62] y [3.63] de uso de cuaternios, un objeto localizado en el sistema de referencia $\{S_i\}$ por su vector de posición a_i y su cuaternio de rotación $\mathbf{R}_i$, tendrá en el sistema de referencia $\{S_{i-1}\}$ el vector de posición $\mathbf{a}_{i-1}$ y el cuaternio $\mathbf{R}_{i-1}$ siguientes[1]:

$$\begin{aligned} (0,\mathbf{a}_{i-1}) &= Q_i(0,\mathbf{a}_i)Q_i^* + (0,\mathbf{p}_i) \\ \mathbf{R}_{i-1} &= Q_i\mathbf{R}_i \end{aligned} \qquad [4.12]$$

donde $\mathbf{p}_i$ y Q_i son respectivamente el desplazamiento y posterior rotación que permiten convertir $\{S_{i-1}\}$ en $\{S_i\}$. Aplicando reiteradamente la expresión anterior a los sistemas de referencia $\{S_0\},\{S_1\},\{S_2\},\{S_3\}$ y $\{S_4\}$ se tendrá:

[1] Por claridad en la exposición, se ha suprimido el símbolo $\circ$ de producto entre cuaterníos.

$$\begin{aligned}
&(0,\mathbf{a}_0) = Q_1(0,\mathbf{a}_1)Q_1^* + (0,\mathbf{p}_1)\\
&\mathbf{R}_0 = Q_1\mathbf{R}_1\\
&(0,\mathbf{a}_1) = Q_2(0,\mathbf{a}_2)Q_2^* + (0,\mathbf{p}_2)\\
&\mathbf{R}_1 = Q_2\mathbf{R}_2\\
&(0,\mathbf{a}_2) = Q_3(0,\mathbf{a}_3)Q_3^* + (0,\mathbf{p}_3)\\
&\mathbf{R}_2 = Q_3\mathbf{R}_3\\
&(0,\mathbf{a}_3) = Q_4(0,\mathbf{a}_4)Q_4^* + (0,\mathbf{p}_4)\\
&\mathbf{R}_3 = Q_4\mathbf{R}_4
\end{aligned} \qquad [4.13]$$

Sustituyendo de manera consecutiva en las expresiones anteriores, se obtiene:

$$\begin{aligned}
(0,\mathbf{a}_0) &= Q_1\Big[Q_2\Big[Q_3\Big[Q_4\big(0,\mathbf{a}_4\big)Q_4^* + \big(0,\mathbf{p}_4\big)\Big]Q_3^* + (0,\mathbf{p}_3)\Big]Q_2^* + (0,\mathbf{p}_2)\Big]Q_1^* + (0,\mathbf{p}_1) =\\
&= Q_1Q_2Q_3Q_4(0,\mathbf{a}_4)Q_4^*Q_3^*Q_2^*Q_1^* + Q_1Q_2Q_3(0,\mathbf{p}_4)Q_3^*Q_2^*Q_1^* +\\
&\quad + Q_1Q_2(0,\mathbf{p}_3)Q_2^*Q_1^* + Q_1(0,\mathbf{p}_2)Q_1^* + (0,\mathbf{p}_1) =\\
&= Q_{1234}(0,\mathbf{a}_4)Q_{1234}^* + Q_{123}(0,\mathbf{p}_4)Q_{123}^* + Q_{12}(0,\mathbf{p}_3)Q_{12}^* + Q_1(0,\mathbf{p}_2)Q_1^* + (0,\mathbf{p}_1)
\end{aligned} \qquad [4.14]$$

donde se ha tenido en cuenta que:

$$\begin{aligned}
Q_{1234} &= Q_1Q_2Q_3Q_4 = (\hat{C}_{124},0,0,\hat{S}_{124})\\
Q_{123} &= Q_1Q_2Q_3 = (\hat{C}_{12},0,0,\hat{S}_{12})\\
Q_{12} &= Q_1Q_2 = (\hat{C}_{12},0,0,\hat{S}_{12})
\end{aligned} \qquad [4.15]$$

y que:

$$Q_{ij}^* = (Q_iQ_j)^* = Q_j^*Q_i^* \qquad [4.16]$$

Desarrollando los productos de cuaternios de la expresión [4.14] se tiene:

$$\begin{aligned}
&Q_{1234}(0,\mathbf{a}_4)Q_{1234}^* = Q_{1234}(0,a_{4x},a_{4y},a_{4z})Q_{1234}^* =\\
&\quad = (-\hat{S}_{124}\,a_{4z},\hat{C}_{124}\,a_{4x} - \hat{S}_{124}\,a_{4y},\hat{C}_{124}\,a_{4y} - \hat{S}_{124}\,a_{4x},\hat{C}_{124}\,a_{4z})Q_{1234}^* =\\
&\quad = (0,\hat{C}_{112244}\,a_{4x} - \hat{S}_{112244}\,a_{4y},\hat{C}_{112244}\,a_{4y} - \hat{S}_{112244}\,a_{4x},a_{4z})\\
&Q_{123}(0,\mathbf{p}_4)Q_{123}^* + Q_{12}(0,\mathbf{p}_3)Q_{12}^* = Q_{12}(0,\mathbf{p}_4 + \mathbf{p}_3)Q_{12}^* = (0,l_3\hat{C}_{1122},l_3\hat{S}_{1122},-q_3)\\
&Q_1(0,\mathbf{p}_2)Q_1^* = (0,l_2\hat{C}_{11},l_2\hat{S}_{11},0)
\end{aligned} \qquad [4.17]$$

Asimismo, según [4.9] :

$$(0,\mathbf{p}_1) = (0,0,0,l_1) \qquad [4.18]$$

con lo que finalmente resulta:

$$\begin{aligned}(0,\mathbf{a}_0) = (0, & a_{4x}\hat{C}_{112244} - a_{4y}\hat{S}_{112244} + l_3\hat{C}_{1122} + l_2\hat{C}_{11}, \\ & a_{4y}\hat{C}_{112244} - a_{4x}\hat{S}_{112244} + l_3\hat{S}_{1122} + l_2\hat{S}_{11}, a_{4z} - q_3 + l_1)\end{aligned} \qquad [4.19]$$

En cuanto a la relación entre los cuaternios que definen la orientación de un objeto en los sistemas $\{S_0\}$ y $\{S_4\}$ se tendrá:

$$\mathbf{R}_0 = Q_1Q_2Q_3Q_4\mathbf{R}_4 = Q_{1234}\mathbf{R}_4 = (\hat{C}_{124},0,0,\hat{S}_{124}) \qquad [4.20]$$

Las expresiones [4.19] y [4.20] permiten conocer la posición $\mathbf{a}_0$ y orientación $\mathbf{R}_0$ de un objeto en el sistema $\{S_0\}$ conocidas éstas en el sistema $\{S_4\}$.

Si, en particular, este objeto está posicionado y orientado en el extremo del robot, se tendrá que:

$$\begin{aligned}\mathbf{a}_4 &= (0,0,0) \\ \mathbf{R}_4 &= (1,0,0,0)\end{aligned} \qquad [4.21]$$

con lo que:

$$\begin{aligned}(0,\mathbf{a}_0) &= (0, l_3\hat{C}_{1122} + l_2\hat{C}_{11}, l_3\hat{S}_{1122} + l_2\hat{S}_{11}, l_1 - q_3) \\ \mathbf{R}_0 &= (\hat{C}_{124},0,0,\hat{S}_{124})\end{aligned} \qquad [4.22]$$

lo que indica que el extremo del robot referido al sistema de su base $\{S_0\}$, está posicionado en:

$$\begin{aligned}x = a_{0x} &= l_3\cos(q_1 + q_2) + l_2\cos q_1 \\ y = a_{0y} &= l_3\mathrm{sen}(q_1 + q_2) + l_2\mathrm{sen}\, q_1 \\ z = a_{0z} &= l_1 - q_3\end{aligned} \qquad [4.23]$$

y está girado respecto al sistema de la base un ángulo $q_1 + q_2 + q_4$ según la rotación en torno al eje **z**:

$$\mathbf{Rot}(\mathbf{z}, q_1 + q_2 + q_4) \qquad [4.24]$$

Las expresiones [4.23] y [4.24] permiten conocer la localización del extremo del robot referidas al sistema de la base en función de las coordenadas articulares (q_1, q_2, q_3, q_4), correspondiendo por lo tanto a la solución del problema cinemático directo.

4.2. CINEMÁTICA INVERSA

El objetivo del problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $q = [q_1, q_2, \ldots, q_n]^T$ para que su extremo se posicione y oriente según una determinada localización espacial.

Así como es posible abordar el problema cinemático directo de una manera sistemática a partir de la utilización de matrices de transformación homogéneas, e independientemente de la configuración del robot, no ocurre lo mismo con el problema cinemático inverso, siendo el procedimiento de obtención de las ecuaciones fuertemente dependiente de la configuración del robot.

Se han desarrollado algunos procedimientos genéricos susceptibles de ser programados [GOLDENBERG-85], de modo que un computador pueda, a partir del conocimiento de la cinemática del robot (con sus parámetros de Denavit-Hartenberg, por ejemplo) obtener la n-upla de valores articulares que posicionan y orientan su extremo. El inconveniente de estos procedimientos es que se trata de métodos numéricos iterativos, cuya velocidad de convergencia e incluso su convergencia en sí no está siempre garantizada.

A la hora de resolver el problema cinemático inverso es mucho más adecuado encontrar una solución cerrada. Esto es, encontrar una relación matemática explícita de la forma:

$$\begin{aligned} q_k &= \mathrm{f}_k(x,y,z,\alpha,\beta,\gamma) \\ \mathrm{k} &= 1\ldots \mathrm{n} \qquad (\mathrm{GDL}) \end{aligned} \qquad [4.25]$$

Este tipo de solución presenta, entre otras, las siguientes ventajas:

1. En muchas aplicaciones, el problema cinemático inverso ha de resolverse en tiempo real (por ejemplo, en el seguimiento de una determinada trayectoria). Una solución de tipo iterativo no garantiza tener la solución en el momento adecuado.
2. Al contrario de lo que ocurría en el problema cinemático directo, con cierta frecuencia la solución del problema cinemático inverso no es única; existiendo diferentes n-uplas $[q_1, \ldots, q_n]^T$ que posicionan y orientan el extremo del robot del mismo modo. En estos casos una solución cerrada permite incluir determinadas reglas o restricciones que aseguren que la solución obtenida sea la más adecuada de entre las posibles (por ejemplo, límites en los recorridos articulares).

No obstante, a pesar de las dificultades comentadas, la mayor parte de los robots poseen cinemáticas relativamente simples que facilitan en cierta medida la resolución de su problema cinemático inverso. Por ejemplo, si se consideran sólo los tres primeros grados de libertad de muchos robots, éstos tienen una

estructura planar, esto es, los tres primeros elementos quedan contenidos en un plano. Esta circunstancia facilita la resolución del problema. Asimismo, en muchos robots se da la circunstancia de que los tres grados de libertad últimos, dedicados fundamentalmente a orientar el extremo del robot, corresponden a giros sobre ejes que se cortan en un punto. De nuevo esta situación facilita el cálculo de la n-upla $[q_1, \ldots, q_n]^T$ correspondiente a la posición y orientación deseadas. Por lo tanto, para los casos citados y otros, es posible establecer ciertas pautas generales que permitan plantear y resolver el problema cinemático inverso de una manera sistemática.

Los métodos geométricos permiten obtener normalmente los valores de las primeras variables articulares, que son las que consiguen posicionar el robot (prescindiendo de la orientación de su extremo). Para ello utilizan relaciones trigonométricas y geométricas sobre los elementos del robot. Se suele recurrir a la resolución de triángulos formados por los elementos y articulaciones del robot.

Como alternativa para resolver el mismo problema se puede recurrir a manipular directamente las ecuaciones correspondientes al problema cinemático directo. Es decir, puesto que éste establece la relación:

$$\begin{bmatrix} \mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} t_{ij} \end{bmatrix} \qquad [4.26]$$

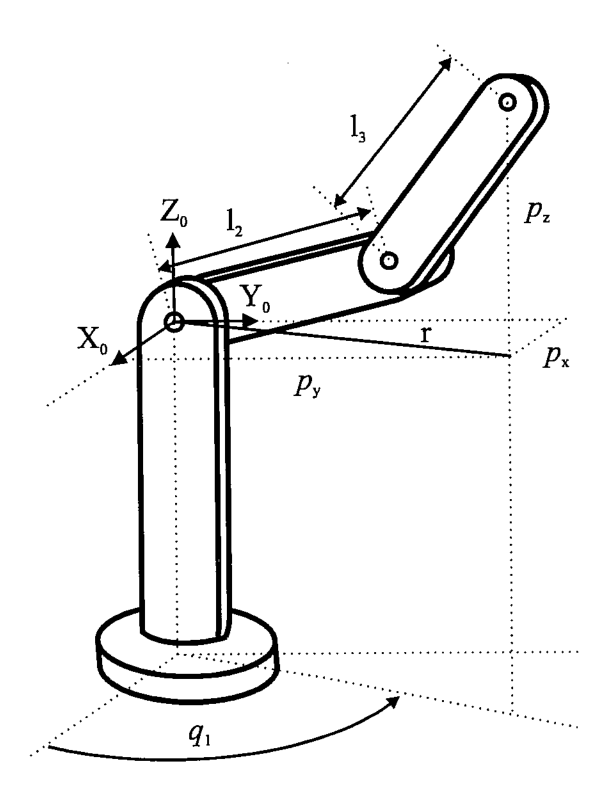


Figura 4.7. Robot articular.

donde los elementos t_{ij} son función de las coordenadas articulares $[q_1, \dots, q_n]^T$, es posible pensar que mediante ciertas combinaciones de las 12 ecuaciones planteadas en [4.26] se puedan despejar las n variables articulares q_i en función de las componentes de los vectores **n**, **o**, **a** y **p**.

Por último, si se consideran robots con capacidad de posicionar y orientar su extremo en el espacio, esto es, robots con 6 GDL, el método de desacoplamiento cinemático permite, para determinados tipos de robots, resolver los primeros grados de libertad, dedicados al posicionamiento, de manera independiente a la resolución de los últimos grados de libertad, dedicados a la orientación. Cada uno de estos dos problemas más simples podrá ser tratado y resuelto por cualquier procedimiento.

4.2.1. Resolución del problema cinemático inverso por métodos geométricos

Como se ha indicado, este procedimiento es adecuado para robots de pocos grados de libertad o para el caso de que se consideren sólo los primeros grados de libertad, dedicados a posicionar el extremo.

El procedimiento en sí se basa en encontrar suficiente número de relaciones geométricas en las que intervendrán las coordenadas del extremo del robot, sus coordenadas articulares y las dimensiones físicas de sus elementos.

Para mostrar el procedimiento a seguir se va a aplicar el método a la resolución del problema cinemático inverso de un robot con 3 GDL de rotación (estructura típica articular). La Figura 4.7 muestra la configuración del robot. El dato de partida son la coordenadas (p_x, p_y, p_z) referidas a $\{S_0\}$ en las que se quiere posicionar su extremo.

Como se ve, este robot posee una estructura planar, quedando este plano definido por el ángulo de la primera variable articular q_1.

El valor de q_1 se obtiene inmediatamente como:

$$q_1 = \operatorname{arctg}\left(\frac{p_y}{p_x}\right) \qquad [4.27]$$

Considerando ahora únicamente los elementos 2 y 3 que están situados en un plano (Figura 4.8-a), y utilizando el teorema del coseno, se tendrá:

$$\left.\begin{array}{l} r^2 = p_x^2 + p_y^2 \\ r^2 + p_z^2 = l_2^2 + l_3^2 + 2 l_2 l_3 \cos q_3 \end{array}\right\} \Rightarrow$$
$$\cos q_3 = \frac{p_x^2 + p_y^2 + p_z^2 - l_2^2 - l_3^2}{2 l_2 l_3} \qquad [4.28]$$

Esta expresión permite obtener q_3 en función del vector de posición del extremo **p**. No obstante, y por motivos de ventajas computacionales, es más conveniente utilizar la expresión de la arcotangente en lugar del arcoseno.

Puesto que

$$\operatorname{sen} q_3 = \pm\sqrt{1-\cos^2 q_3} \qquad [4.29]$$

se tendrá que

$$q_3 = \operatorname{arctg}\left(\frac{\pm\sqrt{1-\cos^2 q_3}}{\cos q_3}\right)$$

$$\text{con} \quad \cos q_3 = \frac{p_x^2 + p_y^2 + p_z^2 - l_2^2 - l_3^2}{2 l_2 l_3} \qquad [4.\ 30]$$

Como se ve, existen 2 posibles soluciones para q_3 según se tome el signo positivo o el signo negativo en la raíz. Éstas corresponden a las configuraciones de codo arriba (Figura 4.8-a) y codo abajo (ver Figura 4.8-b) del robot.

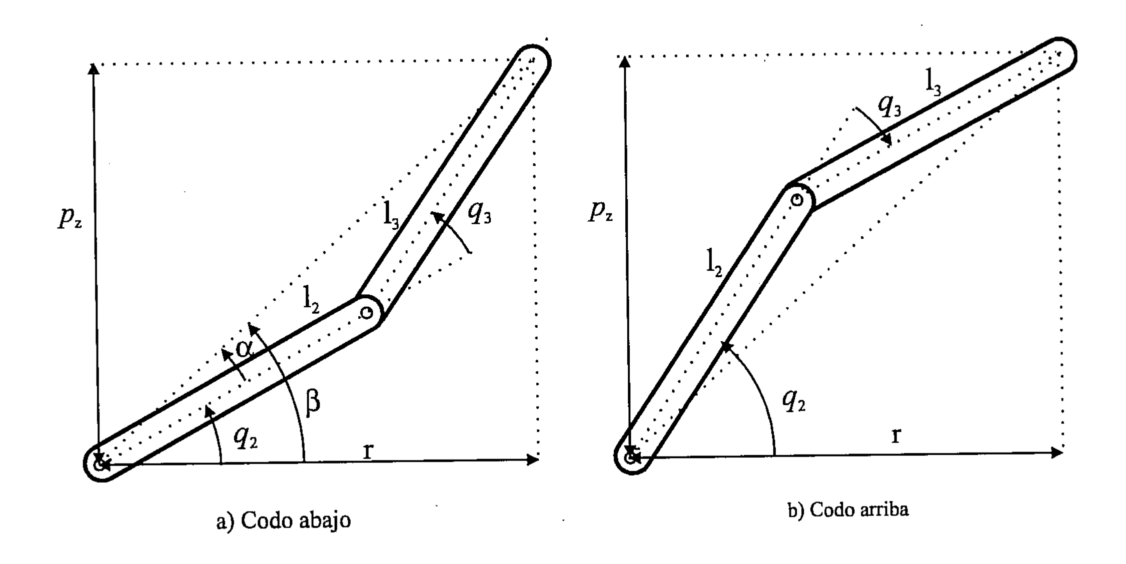


Figura 4.8. Elementos 2 y 3 del robot de la Figura 4.7 contenidos en un plano y en a) configuración codo abajo y b) configuración codo arriba.

El cálculo de q_2 se hace a partir de la diferencia entre β y α:

$$q_2 = \beta - \alpha \qquad [4.31]$$

Siendo:

$$\beta = \operatorname{arctg}\left(\frac{p_z}{r}\right) = \operatorname{arctg}\left(\frac{p_z}{\pm\sqrt{p_x^2 + p_y^2}}\right)$$
$$\alpha = \operatorname{arctg}\left(\frac{l_3 \operatorname{sen} q_3}{l_2 + l_3 \cos q_3}\right) \qquad [4.32]$$

Luego, finalmente

$$q_2 = \operatorname{arctg}\left(\frac{p_z}{\pm\sqrt{p_x^2 + p_y^2}}\right) - \operatorname{arctg}\left(\frac{l_3 \operatorname{sen} q_3}{l_2 + l_3 \cos q_3}\right) \qquad [4.33]$$

De nuevo los dos posibles valores según la elección del signo dan lugar a dos valores diferentes de q_2 correspondientes a las configuraciones codo arriba y abajo.

Las expresiones [4.27], [4.30] y [4.33] resuelven el problema cinemático inverso para el robot de 3 GDL considerado.

4.2.2. Resolución del problema cinemático inverso a partir de la matriz de transformación homogénea

En principio es posible tratar de obtener el modelo cinemático inverso de un robot a partir del conocimiento de su modelo directo. Es decir, suponiendo conocidas las relaciones que expresan el valor de la posición y orientación del extremo del robot en función de sus coordenadas articulares, obtener por manipulación de aquéllas las relaciones inversas.

Sin embargo, en la práctica esta tarea no es trivial siendo en muchas ocasiones tan compleja que obliga a desecharla. Además, puesto que el problema cinemático directo, resuelto a través de la expresión [4.26] contiene en el caso de un robot de 6 GDL 12 ecuaciones, y se buscan sólo 6 relaciones (una por cada grado de libertad), existirán necesariamente ciertas dependencias entre las 12 expresiones de partida (resultado de la condición de ortonormalidad de los vectores **n**, **o** y **a**) con lo cual la elección de qué ecuaciones de [4.26] escoger debe hacerse con sumo cuidado.

Se va a aplicar este procedimiento al robot de 3 GDL de configuración esférica (2 giros y un desplazamiento) mostrado en la Figura 4.9. El robot queda siempre contenido en un plano determinado por el ángulo q_1.

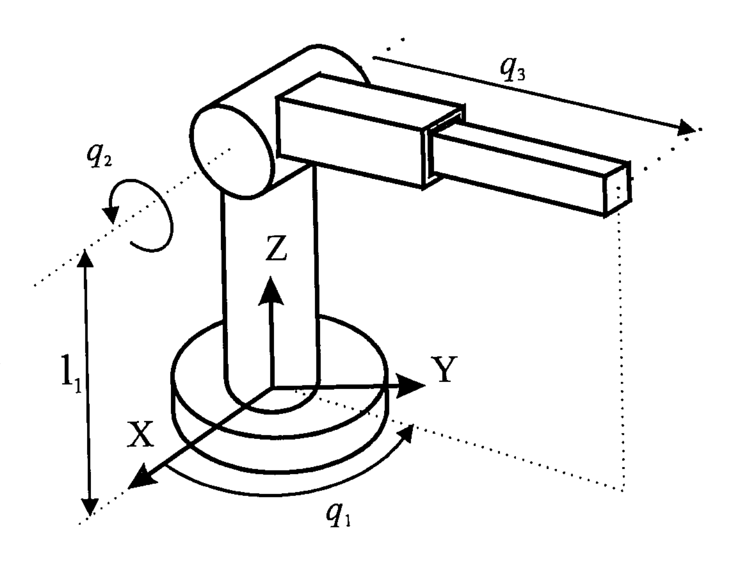


Figura 4.9. Robot polar de 3 GDL.

El primer paso a dar para resolver el problema cinemático inverso es obtener la expresión [4.26] correspondiente a este robot. Es decir, obtener la matriz **T** que relaciona el sistema de referencia $\{S_0\}$ asociado a la base con el sistema de referencia $\{S_3\}$ asociado a su extremo. La Figura 4.10 representa la asignación de sistemas de referencia según los criterios de Denavit-Hartenberg, con el robot situado en su posición de partida ($q_1 = q_2 = 0$), y la Tabla 4.3 muestra los valores de los parámetros de Denavit-Hartenberg.

A partir de éstos es inmediato obtener las matrices **A** y la matriz **T**.

$$
{}^{0}\mathbf{A}_1 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad
{}^{1}\mathbf{A}_2 = \begin{bmatrix} C_2 & 0 & -S_2 & 0 \\ S_2 & 0 & C_2 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad
{}^{2}\mathbf{A}_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

[4. 34]

$$
{}^{0}\mathbf{A}_2 = \begin{bmatrix} C_1C_2 & -S_1 & -C_1S_2 & 0 \\ S_1C_2 & C_1 & -S_1S_2 & 0 \\ S_2 & 0 & C_2 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad
\mathbf{T} = {}^{0}\mathbf{A}_3 = \begin{bmatrix} C_1C_2 & -S_1 & -C_1S_2 & -q_3 C_1S_2 \\ S_1C_2 & C_1 & -S_1S_2 & -q_3 S_1S_2 \\ S_2 & 0 & C_2 & q_3 C_2 + l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Obtenida la expresión de **T** en función de las coordenadas articulares (q_1 , q_2 , q_3), y supuesta una localización de destino para el extremo del robot definida por los vectores **n**, **o**, **a** y **p** se podría intentar manipular directamente las 12 ecuaciones resultantes de **T** a fin de despejar q_1, q_2, y q_3 en función de **n**, **o**, **a** y **p**.

Tabla 4.3. Parámetros D-H del robot de la Figura 4.9.

Articulación	θ	d	a	α
1	q_1	l_1	0	90
2	q_2	0	0	-90
3	0	q_3	0	0

Sin embargo, este procedimiento directo es complicado, apareciendo ecuaciones trascendentes. En lugar de ello, suele ser más adecuado aplicar el siguiente procedimiento:

Puesto que $\mathbf{T} = {}^0\mathbf{A}_1\, {}^1\mathbf{A}_2\, {}^2\mathbf{A}_3$ se tendrá que:

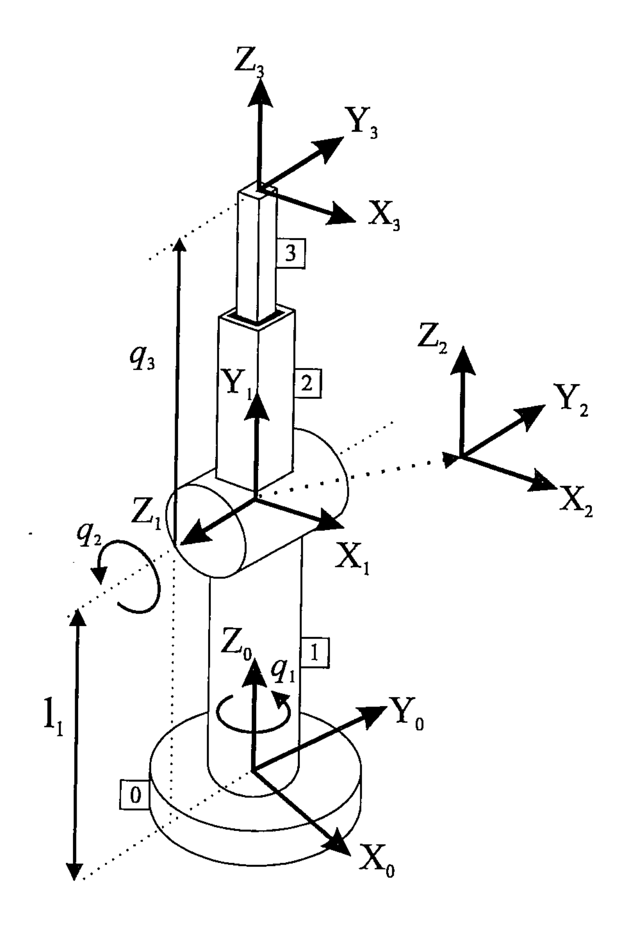


Figura 4.10. Asignación de sistemas de referencia del robot polar de la Figura 4.9.

$$\left({}^{0}\mathbf{A}_1\right)^{-1}\mathbf{T}={}^{1}\mathbf{A}_2\,{}^{2}\mathbf{A}_3$$
$$\left({}^{1}\mathbf{A}_2\right)^{-1}\left({}^{0}\mathbf{A}_1\right)^{-1}\mathbf{T}={}^{2}\mathbf{A}_3 \qquad [4.35]$$

Puesto que $\mathbf{T}=\begin{bmatrix}\mathbf{n} & \mathbf{o} & \mathbf{a} & \mathbf{p}\\ 0 & 0 & 0 & 1\end{bmatrix}$ es conocida, los miembros a la izquierda en las expresiones [4.35] son función de las variables articulares (q_1 , ... , q_k)mientras que los miembros de la derecha lo son de las variables articulares (q_{k+1} , ... , q_n).

De este modo, de la primera de las expresiones de [4.35] se tendrá q_1 aislado del resto de las variables articulares y tal vez será posible obtener su valor sin la complejidad que se tendría abordando directamente la manipulación de la expresión [4.26]. A su vez, una vez obtenida q_1 , la segunda expresión de [4.35] permitirá tener el valor de q_2 aislado respecto de q_3. Por último, conocidos q_1 y q_2 se podrá obtener q_3 de la expresión [4.26] sin excesiva dificultad.

Para poder aplicar este procedimiento, es necesario en primer lugar obtener las inversas de las matrices, ${}^{i-1}\mathbf{A}_i$. Esto es sencillo si se considera que la inversa de una matriz de transformación homogénea viene dada por [3.44]:

$$\begin{bmatrix} n_x & o_x & a_x & p_x\\ n_y & o_y & a_y & p_y\\ n_z & o_z & a_z & p_z\\ 0 & 0 & 0 & 1\end{bmatrix}^{-1}=\begin{bmatrix} n_x & n_y & n_z & -\mathbf{n}^T\mathbf{p}\\ o_x & o_y & o_z & -\mathbf{o}^T\mathbf{p}\\ a_x & a_y & a_z & -\mathbf{a}^T\mathbf{p}\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [4.36]$$

Luego se tiene que:

$${}^{0}\mathbf{A}_1^{-1}=\begin{bmatrix} C_1 & 0 & S_1 & 0\\ S_1 & 0 & -C_1 & 0\\ 0 & 1 & 0 & l_1\\ 0 & 0 & 0 & 1\end{bmatrix}^{-1}=\begin{bmatrix} C_1 & S_1 & 0 & 0\\ 0 & 0 & 1 & -l_1\\ S_1 & -C_1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$${}^{1}\mathbf{A}_2^{-1}=\begin{bmatrix} C_2 & 0 & -S_2 & 0\\ S_2 & 0 & C_2 & 0\\ 0 & -1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}^{-1}=\begin{bmatrix} C_2 & S_2 & 0 & 0\\ 0 & 0 & -1 & 0\\ -S_2 & C_2 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [4.37]$$

$${}^{2}\mathbf{A}_3^{-1}=\begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & q_3\\ 0 & 0 & 0 & 1\end{bmatrix}^{-1}=\begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & -q_3\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Por lo tanto, utilizando la primera de las ecuaciones de [4.35] se tiene que:

$$\left({}^{0}\mathbf{A}_1\right)^{-1}{}^{0}\mathbf{T}_3={}^{1}\mathbf{A}_2\,{}^{2}\mathbf{A}_3=\begin{bmatrix} C_1 & S_1 & 0 & 0\\ 0 & 0 & 1 & -l_1\\ S_1 & -C_1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} n_x & o_x & a_x & p_x\\ n_y & o_y & a_y & p_y\\ n_z & o_z & a_z & p_z\\ 0 & 0 & 0 & 1\end{bmatrix}=$$
$$=\begin{bmatrix} C_2 & 0 & -S_2 & 0\\ S_2 & 0 & C_2 & 0\\ 0 & -1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & q_3\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix} C_2 & 0 & -S_2 & -S_2q_3\\ S_2 & 0 & C_2 & C_2q_3\\ 0 & -1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [4.38]$$

De las 12 relaciones establecidas en la ecuación [4.38] interesan aquellas que expresan q_1 en función de constantes (y no de q_2 y q_3). Así por ejemplo, tomando el elemento (3,4) se tiene:

$$S_1 p_x - C_1 p_y = 0 \quad \Rightarrow$$
$$\tan(q_1) = \left(\frac{p_y}{p_x}\right) \quad \Rightarrow$$

$$q_1 = \arctan\left(\frac{p_y}{p_x}\right) \qquad [4.39]$$

Utilizando ahora la segunda de las ecuaciones de [4.35] se tendrá:

$$\left({}^{1}\mathbf{A}_2\right)^{-1}\left({}^{0}\mathbf{A}_1\right)^{-1}\mathbf{T}={}^{2}\mathbf{A}_3=$$
$$=\begin{bmatrix} C_2 & S_2 & 0 & 0\\ 0 & 0 & -1 & 0\\ -S_2 & C_2 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} C_1 & S_1 & 0 & 0\\ 0 & 0 & 1 & -l_1\\ S_1 & -C_1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} n_x & o_x & a_x & p_x\\ n_y & o_y & a_y & p_y\\ n_z & o_z & a_z & p_z\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & q_3\\ 0 & 0 & 0 & 1\end{bmatrix} \qquad [4.40]$$
$$=\begin{bmatrix} C_2C_1 & C_2S_1 & S_2 & -l_1S_2\\ -S_1 & C_1 & 0 & 0\\ -S_2C_1 & -S_2S_1 & C_2 & -C_2l_1\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix} n_x & o_x & a_x & p_x\\ n_y & o_y & a_y & p_y\\ n_z & o_z & a_z & p_z\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & q_3\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Tomando el elemento (1,4) se tiene:

$$C_2C_1p_x + C_2S_1p_y + S_2p_z - l_1S_2 = 0 \Rightarrow$$
$$C_2\left(C_1p_x + S_1p_y\right) + S_2\left(p_z - l_1\right) = 0 \Rightarrow$$

$$\tan(q_2) = -\frac{C_1p_x + S_1p_y}{\left(p_z - l_1\right)} \qquad [4.41]$$

y considerando que por [4.39] :

$$S_1\,p_x - C_1\,p_y = 0 \Rightarrow$$
$$\left(S_1\,p_x - C_1\,p_y\right)^2 = S_1^2\,p_x^2 + C_1^2\,p_y^2 - 2S_1C_1\,p_xp_y = 0 \Rightarrow$$
$$\left(1 - C_1^2\right)p_x^2 + \left(1 - S_1^2\right)p_y^2 = 2S_1C_1\,p_xp_y \Rightarrow$$
$$C_1^2\,p_x^2 + S_1^2\,p_y^2 + 2S_1C_1\,p_xp_y = p_x^2 + p_y^2 \Rightarrow$$

$$C_1p_x + S_1p_y = \sqrt{p_x^2 + p_y^2} \qquad [4.42]$$

se tiene finalmente:

$$q_2 = \arctan\frac{\sqrt{p_x^2 + p_y^2}}{l_1 - p_z} \qquad [4.43]$$

Por último, tomando de [4.40] el elemento (3,4) se tiene:

$$-S_2C_1p_x - S_2S_1p_y + C_2p_z - C_2l_1 = q_3 \Rightarrow$$
$$C_2(p_z - l_1) - S_2(C_1p_x + S_1p_y) = q_3 \Rightarrow$$

$$q_3 = C_2(p_z - l_1) - S_2\sqrt{p_x^2 + p_y^2} \qquad [4.44]$$

Las expresiones [4.39], [4.43] y [4.44] corresponden a la solución del problema cinemático inverso del robot considerado. A continuación se reproducen estas expresiones.

$$q_1 = \arctan\left(\frac{p_y}{p_x}\right)$$

$$q_2 = \arctan\frac{\sqrt{p_x^2 + p_y^2}}{l_1 - p_z} \qquad [4.45]$$

$$q_3 = C_2(p_z - l_1) - S_2\sqrt{p_x^2 + p_y^2}$$

A los mismos resultados se podría haber llegado mediante consideraciones geométricas.

4.2.3. Desacoplo cinemático

Los procedimientos vistos en los apartados anteriores permiten obtener los valores de las 3 primeras variables articulares del robot, aquellas que posicionan su extremo en unas coordenadas (p_x , p_y , p_z) determinadas, aunque pueden ser igualmente utilizadas para la obtención de las 6 a costa de una mayor complejidad.

Ahora bien, como es sabido, en general no basta con posicionar el extremo del robot en un punto del espacio, sino que casi siempre es preciso también conseguir que la herramienta que aquél porta se oriente de una manera determinada. Para ello, los robots cuentan con otros tres grados de libertad adicionales, situados al final de la cadena cinemática y cuyos ejes, generalmente, se cortan en un punto, que informalmente se denomina *muñeca del robot*. Si bien la variación de estos tres últimos grados de libertad origina un cambio en la posición final del extremo real del robot, su verdadero objetivo es poder orientar la herramienta del robot libremente en el espacio.

El método de desacoplo cinemático saca partido de este hecho, separando ambos problemas: posición y orientación. Para ello, dada una posición y orientación final deseadas, establece las coordenadas del punto de corte de los 3 últimos ejes (*muñeca del robot*) calculándose los valores de las tres primeras variables articulares (q_1,q_2,q_3) que consiguen posicionar este punto. A continuación, a partir de los datos de orientación y de los ya calculados (q_1,q_2,q_3) obtiene los valores del resto de las variables articulares.

Tabla 4.4. Parámetros D-H del robot de la Figura 4.11.

Articulación	θ	d	a	α
1	θ_1	l_1	0	-90
2	θ_2	0	l_2	0
3	θ_3	0	0	90
4	θ_4	l_3	0	-90
5	θ_5	0	0	90
6	θ_6	l_4	0	0

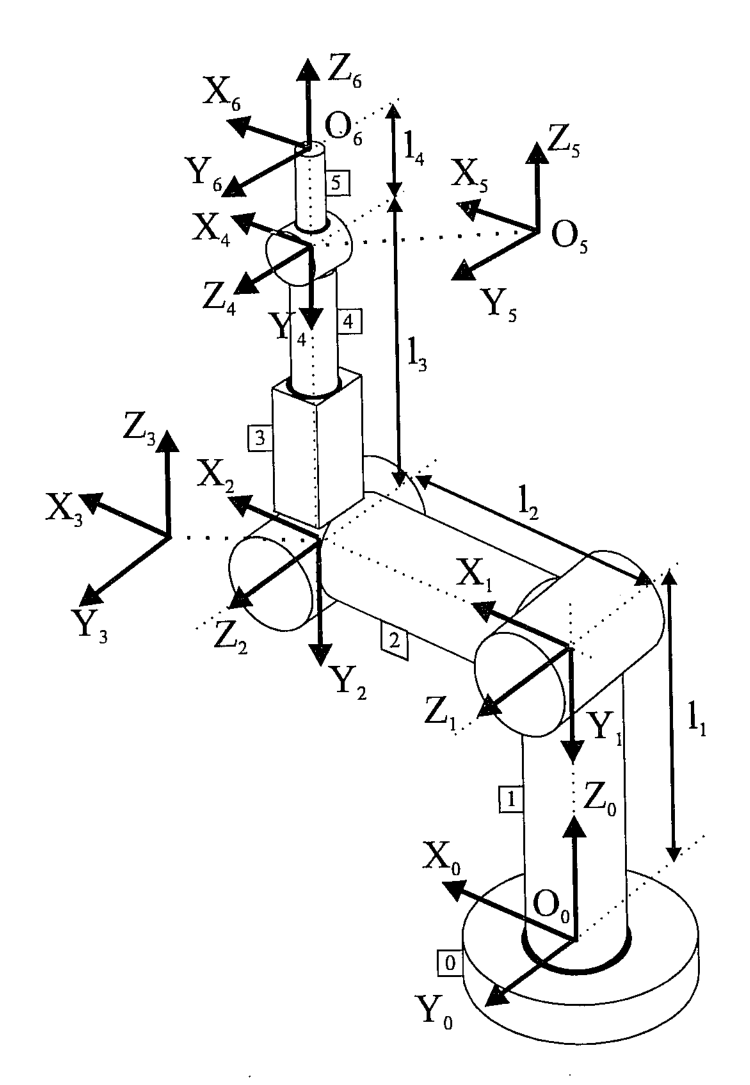


Figura 4.11. Cinemática del robot IRB2400, cuya inversa se puede desacoplar.

En la Figura 4.11 se representa un robot que reúne las citadas características, con indicación de los sistemas de coordenadas asociados según el procedimiento de Denavit-Hartemberg, cuyos parámetros se pueden observar en la Tabla 4.4.

El punto central de la muñeca del robot corresponde al origen del sistema $\{S_5\}$: O_5. Por su parte, el punto final del robot será el origen del sistema $\{S_6\}$: O_6. En lo que sigue se utilizarán los vectores:

$$\begin{aligned} \mathbf{p}_m &= \overline{O_0 O_5} \\ \mathbf{p}_r &= \overline{O_0 O_6} \end{aligned} \qquad [4.46]$$

que van desde el origen del sistema asociado a la base del robot $\{S_0\}$ hasta los puntos centro de la muñeca y fin del robot, respectivamente.

Puesto que la dirección del eje $\mathbf{z}_6$ debe coincidir con la de $\mathbf{z}_5$ y la distancia entre O_5 y O_6 medida a lo largo de $\mathbf{z}_5$ es precisamente $d_4 = l_4$, se tendrá que:

$$\mathbf{p}_m = \mathbf{p}_r - l_4 \mathbf{z}_6 \qquad [4.47]$$

estando todos los vectores referidos a las coordenadas del sistema $\{S_0\}$.

En la expresión [4.46] $\mathbf{p}_r$ son las coordenadas del punto donde se pretende que se posicione el robot expresadas en $\{S_0\}$. Por lo tanto

$$\mathbf{p}_r = \left[p_x, p_y, p_z\right]^T \qquad [4.48]$$

El vector director $\mathbf{z}_6$ es el vector $\mathbf{a}$ correspondiente a la orientación deseada $\mathbf{z}_6 = [a_x, a_y, a_z]^T$ y l_4 es un parámetro asociado con el robot. Por lo tanto, las coordenadas del punto central de la muñeca (p_{mx}, p_{my}, p_{mz}) son fácilmente obtenibles.

Tal y como se mostró en el epígrafe 4.2 es posible, mediante un método geométrico, por ejemplo, calcular los valores de (q_1,q_2,q_3) que consiguen posicionar el robot en el $\mathbf{p}_m$ deseado.

Queda ahora obtener los valores de q_4, q_5, y q_6 que consiguen la orientación deseada. Para ello, denominando ${}^0\mathbf{R}_6$ a la submatriz de rotación de ${}^0\mathbf{T}_6$ se tendrá:

$${}^0\mathbf{R}_6 = [\mathbf{n} \quad \mathbf{o} \quad \mathbf{a}] = {}^0\mathbf{R}_3 \, {}^3\mathbf{R}_6 \qquad [4.49]$$

donde ${}^0\mathbf{R}_6$ es conocida por ser la orientación deseada del extremo del robot, y ${}^0\mathbf{R}_3$ definida por:

$${}^0\mathbf{R}_3 = {}^0\mathbf{A}_1 \, {}^1\mathbf{A}_2 \, {}^2\mathbf{A}_3 \qquad [4.50]$$

también lo será a partir de los valores ya obtenidos de q_1, q_2, y q_3. Por lo tanto:

$${}^3\mathbf{R}_6 = \left[r_{ij}\right] = \left({}^0\mathbf{R}_3\right)^{-1} {}^0\mathbf{R}_6 = \left({}^0\mathbf{R}\right)^T [\mathbf{n} \quad \mathbf{o} \quad \mathbf{a}] \qquad [4.51]$$

tendrá sus componentes numéricas conocidas.

Por otra parte, ${}^3\mathbf{R}_6$ corresponde con la submatriz (3x3) de rotación de la matriz de transformación homogénea ${}^3\mathbf{T}_6$ que relaciona el sistema $\{S_3\}$ con el $\{S_6\}$. Por lo tanto:

$${}^3\mathbf{R}_6 = {}^3\mathbf{R}_4 \, {}^4\mathbf{R}_5 \, {}^5\mathbf{R}_6 \qquad [4.52]$$

donde ${}^{i-1}\mathbf{R}_i$ es la submatriz de rotación de la matriz de Denavit-Hartemberg ${}^{i-1}\mathbf{A}_i$, cuyos valores son:

$$
{}^{3}\mathbf{R}_4 = \begin{bmatrix} C_4 & 0 & -S_4 \\ S_4 & 0 & C_4 \\ 0 & -1 & 0 \end{bmatrix} \quad {}^{4}\mathbf{R}_5 = \begin{bmatrix} C_5 & 0 & S_5 \\ S_5 & 0 & -C_5 \\ 0 & 1 & 0 \end{bmatrix} \quad {}^{5}\mathbf{R}_6 = \begin{bmatrix} C_6 & -S_6 & 0 \\ S_6 & C_6 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad [4.53]
$$

Luego se tiene que:

$$
{}^{3}\mathbf{R}_6 = \begin{bmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & -S_4C_5 \\ -S_5C_6 & S_5S_6 & C_5 \end{bmatrix} \qquad [4.54]
$$

donde r_{ij} serán por [4.51] valores numéricos conocidos :

$$
\left[r_{ij}\right] = \begin{bmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & -S_4C_5 \\ -S_5C_6 & S_5S_6 & C_5 \end{bmatrix} \qquad [4.55]
$$

De las nueve relaciones expresadas en [4.55] se pueden tomar las correspondientes a r_{13}, r_{23}, r_{33}, r_{31} y r_{32} :

$$
\begin{array}{lll} r_{13} = C_4S_5 & r_{23} = -S_4C_5 & r_{33} = C_5 \\ r_{31} = -S_5C_6 & r_{32} = S_5S_6 & \end{array} \qquad [4.56]
$$

Del conjunto de ecuaciones [4.56] es inmediato obtener los valores de los parámetros articulares (se recomienda convertir todas las funciones trigonométricas inversas en su arcotangente, por ser ésta computacionalmente más robusta):

$$
\begin{aligned} q_4 &= \operatorname{arcsen}\left(\frac{r_{23}}{r_{33}}\right) \\ q_5 &= \arccos\left(r_{33}\right) \\ q_6 &= \arctan\left(-\frac{r_{32}}{r_{31}}\right) \end{aligned} \qquad [4.57]
$$

Esta expresión, junto con las [4.27], [4.30] y [4.33], y teniendo en cuenta que las posiciones de cero son distintas, constituyen la solución completa del problema cinemático inverso del robot articular de la Figura 4.11.

4.3. MATRIZ JACOBIANA

El modelado cinemático de un robot busca las relaciones entre las variables articulares y la posición (expresada normalmente en forma de coordenadas cartesianas) y orientación del extremo del robot. En esta relación no se tienen en cuenta las fuerzas o pares que actúan sobre el robot (actuadores, cargas, fricciones, etc.) y que pueden originar el movimiento del mismo. Sin embargo, sí que debe permitir conocer, además de la relación entre las coordenadas articulares y del extremo, la relación entre sus respectivas derivadas. Así, el sistema de control del robot debe establecer qué velocidades debe imprimir a cada articulación (a través de sus respectivos actuadores) para conseguir que el extremo desarrolle una trayectoria temporal concreta, por ejemplo, una línea recta a velocidad constante.

Para este y otros fines, es de gran utilidad disponer de la relación entre las velocidades de las coordenadas articulares y las de la posición y orientación del extremo del robot. La relación entre ambos vectores de velocidad se obtiene a través de la denominada **matriz jacobiana** (Figura 4.12).

La matriz Jacobiana directa permite conocer las velocidades del extremo del robot a partir de los valores de las velocidades de cada articulación. Por su parte, la matriz Jacobiana inversa permitirá conocer las velocidades articulares necesarias para obtener unas velocidades determinadas en el extremo del robot.

4.3.1. Relaciones diferenciales

El método más directo para obtener la relación entre velocidades articulares y del extremo del robot consiste en diferenciar las ecuaciones correspondientes al modelo cinemático directo.

Así, supónganse conocidas las ecuaciones que resuelven el problema cinemático directo de un robot de n GDL:

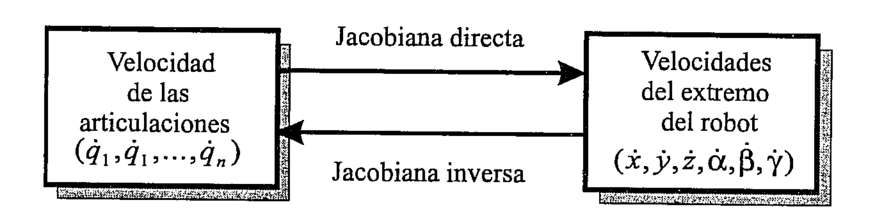


Figura 4.12. Matriz jacobiana directa e inversa.

$$x = f_x(q_1,\ldots,q_n) \qquad y = f_y(q_1,\ldots,q_n) \qquad z = f_z(q_1,\ldots,q_n)$$
$$\alpha = f_\alpha(q_1,\ldots,q_n) \qquad \beta = f_\beta(q_1,\ldots,q_n) \qquad \gamma = f_\gamma(q_1,\ldots,q_n) \qquad [4.58]$$

Si se derivan con respecto al tiempo ambos miembros del conjunto de ecuaciones anteriores, se tendrá:

$$\dot{x} = \sum_1^n \frac{\partial f_x}{\partial q_i}\dot{q}_i \qquad \dot{y} = \sum_1^n \frac{\partial f_y}{\partial q_i}\dot{q}_i \qquad \dot{z} = \sum_1^n \frac{\partial f_z}{\partial q_i}\dot{q}_i$$
$$\dot{\alpha} = \sum_1^n \frac{\partial f_\alpha}{\partial q_i}\dot{q}_i \qquad \dot{\beta} = \sum_1^n \frac{\partial f_\beta}{\partial q_i}\dot{q}_i \qquad \dot{\gamma} = \sum_1^n \frac{\partial f_\gamma}{\partial q_i}\dot{q}_i \qquad [4.59]$$

O expresado en forma matricial:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \mathbf{J} \cdot \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \vdots \\ \dot{q}_n \end{bmatrix} \qquad \text{con} \quad \mathbf{J} = \begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \cdots & \frac{\partial f_x}{\partial q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_\gamma}{\partial q_1} & \cdots & \frac{\partial f_\gamma}{\partial q_n} \end{bmatrix} \qquad [4.60]$$

La matriz **J** se denomina matriz Jacobiana.

Puesto que el valor numérico de cada uno de los elementos $[j_{pq}]$ de la Jacobiana dependerá de los valores instantáneos de las coordenadas articulares θ_i, el valor de la Jacobiana será diferente en cada uno de los puntos del espacio articular.

Ejemplo 4.3.

Se va a obtener la matriz Jacobiana del robot SCARA de la Figura 4.6. El problema cinemático directo viene determinado por las ecuaciones:

$$x = l_3 C_{12} + l_2 C_1$$
$$y = l_3 S_{12} + l_2 S_1$$
$$z = l_1 - q_3$$

Por lo tanto se tendrá:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -\left(l_3S_{12} + l_2S_1\right) & -l_3S_{12} & 0 \\ l_3C_{12} + l_2C_1 & l_3C_{12} & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix}$$

Si el robot se encuentra en un momento determinado en la posición dada por

$$q_1 = \frac{\pi}{6}\text{rad.} \qquad q_2 = \frac{\pi}{4}\text{rad.} \qquad q_3 = 0{,}75\text{m.}$$

moviéndose a una velocidad articular de valor instantáneo:

$$\dot{q}_1 = \frac{\pi}{2}\text{rad / s.} \quad \dot{q}_2 = \frac{\pi}{2}\text{rad / s.} \quad \dot{q}_3 = 1\text{m / s}$$

y con $l_2 = l_3 = 1$ m, la velocidad de su extremo será:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -1{,}465 & -0{,}965 & 0 \\ 1{,}124 & 0.258 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \pi/2 \\ \pi/2 \\ 1 \end{bmatrix} = \begin{bmatrix} -3.81 \\ 2{,}17 \\ -1 \end{bmatrix}$$

Si, manteniéndose la velocidad articular, el robot se encontrase en la posición $q_1=\pi/3$ rad., $q_2=\pi/2$ rad., la velocidad del extremo sería:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -1{,}36 & -0{,}5 & 0 \\ -0{,}366 & -0{,}866 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \pi/2 \\ \pi/2 \\ 1 \end{bmatrix} = \begin{bmatrix} -2{,}92 \\ -1{,}935 \\ -1 \end{bmatrix}$$

4.3.2. Jacobiana inversa

Del mismo modo que se ha obtenido la relación directa que permite obtener las velocidades del extremo a partir de las velocidades articulares, puede obtenerse la relación inversa que permite calcular las velocidades articulares partiendo de las del extremo. En la obtención de la relación inversa pueden emplearse diferentes procedimientos.

En primer lugar, supuesta conocida la relación directa, dada por la matriz Jacobiana [4.60], se puede obtener la relación inversa invirtiendo simbólicamente la matriz

$$\begin{bmatrix} \dot{q}_1 \\ \vdots \\ \vdots \\ \dot{q}_n \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \qquad [4.61]$$

Esta alternativa, de planteamiento sencillo, es en la práctica de difícil realización. Suponiendo que la matriz **J** sea cuadrada, la inversión simbólica de una matriz 6 x 6, cuyos elementos son funciones trigonométricas, es de gran complejidad, siendo este procedimiento inviable.

Como segunda alternativa puede plantearse la evaluación numérica de la matriz **J** para una configuración (q_i) concreta del robot, e invirtiendo numéricamente esta matriz encontrar la relación inversa válida para esa configuración. En este caso hay que considerar, en primer lugar, que el valor numérico de la Jacobiana va cambiando a medida que el robot se mueve y, por lo tanto, la Jacobiana inversa ha de ser recalculada constantemente. Además, pueden existir n-uplas (q_1,...,q_n) para las cuales la matriz Jacobiana **J** no sea invertible por ser su determinante, denominado Jacobiano, nulo. Estas configuraciones del robot en las que el Jacobiano se anula se denominan configuraciones singulares y serán tratadas más adelante. Una tercera dificultad que puede surgir con este y otros procedimientos de computo de la matriz Jacobiana inversa, se deriva de la circunstancia de que la matriz **J** no sea cuadrada. Esto ocurre cuando el número de grados de libertad del robot no coincide con la dimensión del espacio de la tarea (normalmente seis). En el caso de que el número de grados de libertad sea inferior, la matriz Jacobiana tendrá más filas que columnas. Esto quiere decir que el movimiento del robot está sometido a ciertas restricciones (por ejemplo, no se puede alcanzar cualquier orientación). Típicamente esto ocurre en los casos en los que esta restricción no tiene importancia, como en robots dedicados a tareas como soldadura por arco o desbarbado, en las que la orientación de la herramienta en cuanto a su giro en torno al vector **a** es indiferente, por lo que puede ser eliminado este grado de libertad del espacio de la tarea, quedando una nueva matriz Jacobiana cuadrada.

En los casos en que el robot sea redundante (más de 6 GDL o más columnas que filas en la matriz Jacobiana) existirán grados de libertad articulares innecesarios, es decir, que no será preciso mover para alcanzar las nuevas posiciones y velocidades del extremo requeridas. Por ello, la correspondiente velocidad articular podrá ser tomada como cero, o si fuera útil, como un valor constante.

En general, en el caso de que la Jacobiana no sea cuadrada podrá ser usado algún tipo de matriz pseudoinversa, como por ejemplo $(\mathbf{J}\,\mathbf{J}^T)^{-1}$.

La tercera alternativa para obtener la matriz Jacobiana inversa es repetir el procedimiento seguido para la obtención de la Jacobiana directa, pero ahora partiendo del modelo cinemático inverso. Esto es, conocida la relación:

$$
\begin{aligned}
q_1 &= \mathrm{f}_1(x,y,z,\alpha,\beta,\gamma) \\
&\vdots \\
q_n &= \mathrm{f}_n(x,y,z,\alpha,\beta,\gamma)
\end{aligned}
\qquad [4.62]
$$

La matriz jacobiana inversa se obtendrá por diferenciación con respecto del tiempo de ambos miembros de la igualdad:

$$
\begin{bmatrix} \dot{q}_1 \\ \vdots \\ \vdots \\ \vdots \\ \dot{q}_n \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dot{x} \\ \vdots \\ \vdots \\ \vdots \\ \dot{\gamma} \end{bmatrix} \qquad [4.63]
$$

con:

$$
\mathbf{J}^{-1} = \begin{bmatrix}
\dfrac{\partial \mathrm{f}_1}{\partial x} & \cdots & \cdots & \cdots & \dfrac{\partial \mathrm{f}_1}{\partial \gamma} \\
\vdots & \ddots & & & \vdots \\
\vdots & & \ddots & & \vdots \\
\vdots & & & \ddots & \vdots \\
\dfrac{\partial \mathrm{f}_n}{\partial x} & \cdots & \cdots & \cdots & \dfrac{\partial \mathrm{f}_n}{\partial \gamma}
\end{bmatrix} \qquad [4.64]
$$

Como en el caso de la primera alternativa, este método puede ser algebraicamente complicado. Dada la importancia que para el control del movimiento del robot tiene la Jacobiana, se han desarrollado otros procedimientos numéricos para el cálculo rápido de la Jacobiana.

4.3.3. Configuraciones singulares

Se denominan configuraciones singulares de un robot a aquellas en las que el determinante de su matriz Jacobiana (Jacobiano) se anula. Por esta circunstancia, en las configuraciones singulares no existe Jacobiana inversa.

Al anularse el Jacobiano, un incremento infinitesimal de las coordenadas cartesianas supondría un incremento infinito de las coordenadas articulares, lo que en la práctica se traduce en que en las inmediaciones de las configuraciones singulares, el pretender que el extremo del robot se mueva a velocidad constante, obligaría a movimientos de las articulaciones a velocidades inabordables por sus actuadores.

Por ello, en las inmediaciones de las configuraciones singulares se pierde alguno de los grados de libertad del robot, siendo imposible que su extremo se mueva en una determinada dirección cartesiana. Las diferentes configuraciones singulares del robot pueden ser clasificadas como:

- Singularidades en los límites del espacio de trabajo del robot. Se presentan cuando el extremo del robot está en algún punto del límite de trabajo interior o exterior. En esta situación resulta obvio que el robot no podrá desplazarse en las direcciones que lo alejan de este espacio de trabajo.
- Singularidades en el interior del espacio de trabajo del robot. Ocurren dentro de la zona de trabajo y se producen generalmente por el alineamiento de dos o más ejes de las articulaciones del robot.

Ejemplo 4.4.

Para el robot SCARA del que se obtuvo la matriz Jacobiana en el Ejemplo 4.3, se tiene que:

$$\mathbf{J} = \begin{bmatrix} -\left(l_3 S_{12} + l_2 S_1\right) & -l_3 S_{12} & 0 \\ l_3 C_{12} + l_2 C_1 & l_3 C_{12} & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

por lo que el Jacobiano será:

$$|\mathbf{J}| = -\left[-l_3 C_{12}\left(l_3 S_{12} + l_2 S_1\right) + l_3 S_{12}\left(l_3 C_{12} + l_2 C_1\right)\right]$$

que se anula para:

$$l_3 C_{12}\left(l_3 S_{12} + l_2 S_1\right) = l_3 S_{12}\left(l_3 C_{12} + l_2 C_1\right)$$

lo que se cumple siempre que $q_2 = 0$ o π, pues entonces la igualdad anterior se verifica para cualquier q_1:

$$l_3 C_1\left(l_3 S_1 + l_2 S_1\right) = l_3 S_1\left(l_3 C_1 + l_2 C_1\right)$$

Esta situación definida por $q_2 = 0$ o π corresponde a los puntos límite del espacio de trabajo del robot:

- $q_2 = 0$: Límite exterior del espacio de trabajo.
- $q_2 = \pi$: Límite interior del espacio de trabajo.

Se debe prestar especial atención a la localización de las configuraciones singulares del robot para que sean tenidas en cuenta en su control, evitándose solicitar a los actuadores movimientos a velocidades inabordables o cambios bruscos de las mismas.

La Figura 4.13 muestra el resultado de intentar realizar con un robot tipo SCARA, una trayectoria en línea recta a velocidad constante que pasa por una configuración singular. Obsérvese la brusca variación de la velocidad articular $\dot{q}_i$ que crece hasta valores inalcanzables en la práctica.

Para evitar la aparición de configuraciones singulares debe considerarse su existencia desde la propia fase de diseño mecánico, imponiendo restricciones al movimiento del robot o utilizando robots redundantes (lo que conlleva otro tipo de problemas). Finalmente, el sistema de control debe detectar y tratar estas configuraciones evitando pasar precisamente por ellas.

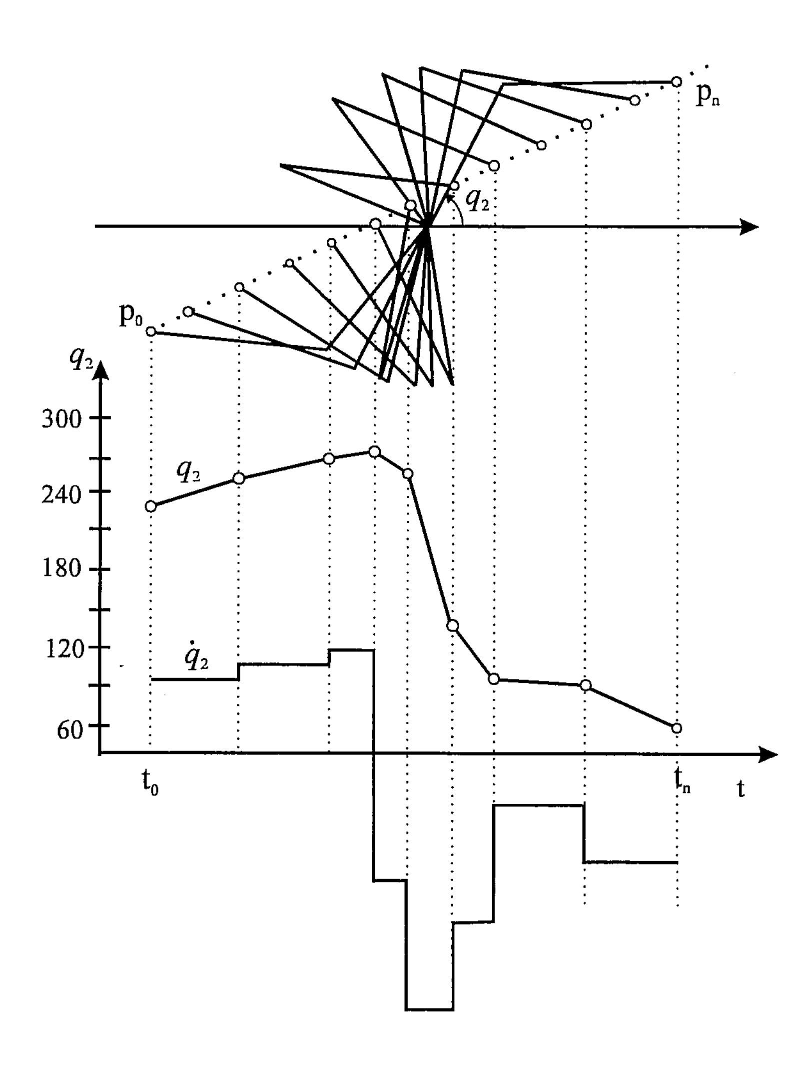


Figura 4.13. Ejemplo de punto singular para un robot tipo SCARA.

Un posible procedimiento para resolver la presencia de una singularidad interior al espacio de trabajo, en la que se pierde la utilidad de alguna articulación (pérdida de algún grado de libertad) sería el siguiente:

1. Identificar la articulación correspondiente al grado de libertad perdido (causante de que el determinante se anule).
2. Eliminar la fila de la Jacobiana correspondiente al grado de libertad perdido y la columna correspondiente a la articulación causante.
3. Con la nueva Jacobiana reducida (rango n-1) obtener las velocidades de todas las articulaciones, a excepción de la eliminada, necesarias para conseguir las velocidades cartesianas deseadas. La velocidad de la articulación eliminada se mantendrá a cero.

4.4. BIBLIOGRAFÍA

[DENAVIT-55] J.Denavit y R.S Hartenberg, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices", *Journal of Applied Mechanics*, junio 1995.

[FU-88] K.S. Fu et. al, "Cinemática del brazo del robot", en: *Robótica. Control, detección, visión e inteligencia*, McGraw-Hill, Madrid, 1988.

[GOLDENBERG-85] A.A. Goldenberg, b. Benhabib y R.G. Fenton, "A complete generalized solution to the inverse kinematics of robots", *IEEE Journal of Robotics and Automation*, RA-1, 1, 1985.

[PAUL-81] R.P Paul, *Robot Manipulators : Mathematics, Programming and Control*, The MIT Press, Massachusetts, 1981.

Capítulo 5

DINÁMICA DEL ROBOT

Díjole el perro al hueso: si tu eres duro, yo tengo tiempo.
Refranero popular

La dinámica se ocupa de la relación entre las fuerzas que actúan sobre un cuerpo y el movimiento que en él se origina. Por lo tanto, el modelo dinámico de un robot tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo.

Esta relación se obtiene mediante el denominado **modelo dinámico**, que relaciona matemáticamente:

1. La localización del robot definida por sus variables articulares o por las coordenadas de localización de su extremo, y sus derivadas: velocidad y aceleración.
2. Las fuerzas y pares aplicados en las articulaciones (o en el extremo del robot).
3. Los parámetros dimensionales del robot, como longitud, masas e inercias de sus elementos.

La obtención de este modelo para mecanismos de uno o dos grados de libertad no es excesivamente compleja, pero a medida que el número de grados de libertad aumenta, el planteamiento y obtención del modelo dinámico se complica enormemente. Por este motivo no siempre es posible obtener un modelo dinámico expresado de una forma cerrada, esto es, mediante una serie de ecuaciones, normalmente de tipo diferencial de 2º orden, cuya integración permita conocer qué movimiento surge al aplicar unas fuerzas o qué fuerzas hay que aplicar para obtener un movimiento determinado. El modelo dinámico debe ser resuelto entonces de manera iterativa mediante la utilización de un procedimiento numérico.

El problema de la obtención del modelo dinámico de un robot es, por lo tanto, uno de los aspectos más complejos de la robótica, lo que ha llevado a ser obviado en numerosas ocasiones. Sin embargo, el modelo dinámico es imprescindible para conseguir los siguientes fines [SALTAREN-96]:

1. Simulación del movimiento del robot.
2. Diseño y evaluación de la estructura mecánica del robot.
3. Dimensionamiento de los actuadores.
4. Diseño y evaluación del control dinámico del robot.

Este último fin es evidentemente de gran importancia, pues de la calidad del control dinámico del robot depende la precisión y velocidad de sus movimientos. La gran complejidad ya comentada existente en la obtención del modelo dinámico del robot, ha motivado que se realicen ciertas simplificaciones, de manera que así pueda ser utilizado en el diseño del controlador.

Es importante hacer notar que el modelo dinámico completo de un robot debe incluir no sólo la dinámica de sus elementos (barras o eslabones) sino también la propia de sus sistemas de transmisión, de los actuadores y sus equipos electrónicos de mando. Estos elementos incorporan al modelo dinámico nuevas inercias, rozamientos, saturaciones de los circuitos electrónicos, etc. aumentando aún más su complejidad.

Por último, es preciso señalar que si bien en la mayor parte de las aplicaciones reales de la robótica, las cargas e inercias manejadas no son suficientes como para originar deformaciones en los eslabones del robot, en determinadas ocasiones no ocurre así, siendo preciso considerar al robot como un conjunto de eslabones no rígidos. Aplicaciones de este tipo pueden encontrarse en la robótica espacial o en robots de grandes dimensiones, entre otras. Un estudio del modelado de robots con estas características puede encontrarse en [AMARIS-95].

Este capítulo va a presentar especial atención a la obtención del modelo dinámico del mecanismo robot más habitual, en el que se considera a éste como rígido, realizándose posteriormente algunas consideraciones sobre la dinámica de los actuadores más habituales en robótica.

5.1. MODELO DINÁMICO DE LA ESTRUCTURA MECÁNICA DE UN ROBOT RÍGIDO

La obtención del modelo dinámico de un mecanismo, y en particular de un robot, se basa fundamentalmente en el planteamiento del equilibrio de fuerzas establecido en la segunda ley de Newton, o su equivalente para movimientos de rotación, la denominada ley de Euler:

$$\sum \mathbf{F} = m\dot{v} \qquad \sum \mathbf{T} = \mathbf{I}\,\dot{\omega} + \omega \times (\mathbf{I}\,\omega) \tag{5. 1}$$

Así, en el caso simple de un robot monoarticular como el representado en la Figura 5.1, el equilibrio de fuerzas-pares daría como resultado la ecuación:

$$\tau = I\frac{d^2\theta}{dt^2} + MgL\cos\theta = ML^2\ddot{\theta} + MgL\cos\theta \qquad [5.2]$$

En donde se ha supuesto que toda la masa se encuentra concentrada en el centro de gravedad del elemento, que no existe rozamiento alguno y que no se manipula ninguna carga.

Para un par motor τ determinado, la integración de la ecuación [5.2] daría lugar a la expresión de $\theta(t)$ y de sus derivadas $\dot{\theta}(t)$ y $\ddot{\theta}(t)$, con lo que sería posible conocer la evolución de la coordenada articular del robot y de su velocidad y aceleración.

De forma inversa, si se pretende que $\theta(t)$ evolucione según una determinada función del tiempo, sustituyendo en [5.2] podría obtenerse el par $\tau(t)$ que sería necesario aplicar. Si el robot tuviese que ejercer alguna fuerza en su extremo, ya sea al manipular una carga o, por ejemplo, realizar un proceso sobre alguna pieza, bastaría con incluir esta condición en la ecuación [5.2] y proceder del mismo modo.

Se tiene así que del planteamiento del equilibrio de fuerzas y pares que intervienen sobre el robot se obtienen los denominados modelos dinámicos directo e inverso:

- **Modelo dinámico directo**: expresa la evolución temporal de las coordenadas articulares del robot en función de las fuerzas y pares que intervienen.
- **Modelo dinámico inverso**: expresa las fuerzas y pares que intervienen en función de la evolución de las coordenadas articulares y sus derivadas.

El planteamiento del equilibrio de fuerzas en un robot real de 5 o 6 grados de libertad, es mucho más complicado que el ejemplo de la Figura 5.1. Debe tenerse en cuenta que junto con las fuerzas de inercia y gravedad, aparecen fuerzas de Coriolis debidas al movimiento relativo existente entre los diversos elementos, así como de fuerzas centrípetas que dependen de la configuración instantánea del manipulador.

Como planteamiento alternativo para la obtención del modelo se puede usar la formulación Lagrangiana, basada en consideraciones energéticas. Este planteamiento es más sistemático que el ante-

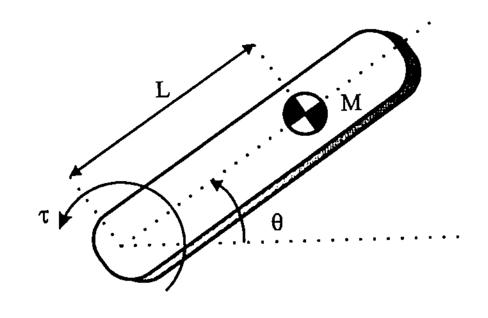


Figura 5.1. Modelo de eslabón con masa concentrada.

rior, y por lo tanto, facilita enormemente la formulación de un modelo tan complejo como el de un robot.

La formulación Lagrangiana establece la ecuación:

$$\frac{d}{dt}\frac{\partial \mathscr{L}}{\partial \dot{q}_i} - \frac{\partial \mathscr{L}}{\partial q_i} = \tau$$
$$\mathscr{L} = k - u \qquad [5.\ 3]$$

con

- q_i: coordenadas generalizadas (en este caso las articulares).
- τ: vector de fuerzas y pares aplicados en las q_i.
- $\mathscr{L}$: Función Lagrangiana.
- k: energía cinética.
- u: energía potencial.

En el caso del robot monoarticular de la Figura 5.1 se tendría:

$$k = \frac{1}{2} I \dot{\theta}^2 \qquad [5.\ 4]$$

donde $I = ML^2$.

Además:

$$u = Mgh = MgLsen\theta \qquad [5.\ 5]$$

luego:

$$\mathscr{L} = k - u = \frac{1}{2} ML^2\dot{\theta}^2 - MgLsen\theta \qquad [5.\ 6]$$

$$\frac{\partial \mathscr{L}}{\partial \theta} = -MgL\cos\theta$$
$$\frac{\partial \mathscr{L}}{\partial \dot{\theta}} = ML^2\dot{\theta} \qquad [5.\ 7]$$
$$\frac{d}{dt}\frac{\partial \mathscr{L}}{\partial \dot{\theta}} = ML^2\ddot{\theta}$$

y sustituyendo en [5.3] se obtiene:

$$ML^2\ddot{\theta} + MgL\cos\theta = \tau \qquad [5.\ 8]$$

ecuación que coincide con la [5.2].

Aunque, para el caso simple del ejemplo, la obtención del modelo mediante la formulación Lagrangiana ha resultado más tediosa que mediante la formulación Newtoniana, la primera muestra sus ventajas a medida que aumenta el número de grados de libertad.

La obtención del modelo dinámico de un robot ha sido y es objeto de estudio e investigación. Numerosos investigadores han desarrollado formulaciones alternativas, basadas fundamentalmente en la mecánica Newtoniana y Lagrangiana, con el objeto de obtener modelos manejables por los sistemas de cálculo de una manera más eficiente.

Algunos de estos planteamientos son los debidos a:

- Uicker-1965: basado en la formulación Lagrangiana.
- Lu-1980: basado en la formulación Newtoniana.
- Lee-1983.

Un estudio completo de estos y otros planteamientos puede encontrarse en [FU-88], [PAUL-81] o [CRAIG-89] entre otros. Aquí se expondrán únicamente los resultados finales de estos planteamientos, que expresados en forma de algoritmo, permiten obtener el modelo dinámico del robot. En ambos casos se aplica la metodología a seguir sobre un robot de 2 grados de libertad.

5.2. OBTENCIÓN DEL MODELO DINÁMICO DE UN ROBOT MEDIANTE LA FORMULACIÓN DE LAGRANGE-EULER

Uicker en 1965 [UICKER-65] [UICKER-64] utilizó la representación de D-H basada en las matrices de transformación homogénea para formular el modelo dinámico de un robot mediante la ecuación de Lagrange.

Este planteamiento utiliza, por tanto, las matrices ${}^{i-1}\mathbf{A}_i$ que relacionan el sistema de coordenadas de referencia del elemento i con el del elemento i-1. Se realizan en este caso operaciones de producto y suma innecesarias (recuérdese la información redundante contenida en las matrices **A** debido a la ortonormalidad de la submatriz de rotación). Se trata de un procedimiento ineficiente desde el punto de vista computacional. Puede comprobarse que el algoritmo es de un orden de complejidad computacional $O(n^4)$, es decir, el número de operaciones a realizar crece con la potencia 4 del número de grados de libertad. Sin embargo, conduce a unas ecuaciones finales bien estructuradas donde aparecen de manera clara los diversos pares y fuerzas que intervienen en el movimiento (inercia, Coriolis, gravedad).

Se presenta a continuación al algoritmo a seguir para obtener el modelo dinámico del robot por el procedimiento de Lagrange-Euler (L-E).

5.2.1. Algoritmo computacional para el modelado dinámico por Lagrange-Euler

L-E 1. Asignar a cada eslabón un sistema de referencia de acuerdo a las normas de D-H.

L-E 2. Obtener las matrices de transformación ${}^0\mathbf{A}_i$ para cada elemento i.

L-E 3. Obtener las matrices $\mathbf{U}_{ij}$ definidas por:

$$\mathbf{U}_{ij} = \frac{\partial\, {}^0\mathbf{A}_i}{\partial q_j} \qquad \text{(ver nota 1)} \qquad [5.\,9]$$

L-E 4. Obtener las matrices $\mathbf{U}_{ijk}$ definidas por :

$$\mathbf{U}_{ijk} = \frac{\partial \mathbf{U}_{ij}}{\partial q_k} \qquad \text{(ver nota 2)} \qquad [5.\,10]$$

L-E 5. Obtener las matrices de pseudoinercias $\mathbf{J}_i$ para cada elemento, que vienen definidas por:

$$\mathbf{J}_i = \begin{bmatrix} \int x_i^2 \mathrm{dm} & \int x_i y_i \mathrm{dm} & \int x_i z_i \mathrm{dm} & \int x_i \mathrm{dm} \\ \int y_i x_i \mathrm{dm} & \int y_i^2 \mathrm{dm} & \int y_i z_i \mathrm{dm} & \int y_i \mathrm{dm} \\ \int z_i x_i \mathrm{dm} & \int z_i y_i \mathrm{dm} & \int z_i^2 \mathrm{dm} & \int z_i \mathrm{dm} \\ \int x_i \mathrm{dm} & \int y_i \mathrm{dm} & \int z_i \mathrm{dm} & \int \mathrm{dm} \end{bmatrix} \qquad [5.\,11]$$

donde las integrales están extendidas al elemento i considerado, y (x_i y_i z_i) son las coordenadas del diferencial de masa dm respecto al sistema de coordenadas del elemento.

L-E 6. Obtener la matriz de inercias $\mathbf{D} = [d_{ij}]$ cuyos elementos vienen definidos por:

$$d_{ij} = \sum_{k=(\max i,j)}^{n} \mathrm{Traza}\left(\mathbf{U}_{kj}\mathbf{J}_k\mathbf{U}_{ki}^T\right) \qquad [5.\,12]$$

con $\quad$ i,j = 1,2,...,n

n: número de grados de libertad

L-E 7. Obtener los términos h_{ikm} definidos por:

$$h_{ikm} = \sum_{j=\max(i,k,m)}^{n} \text{Traza}\left(\mathbf{U}_{jkm}\mathbf{J}_j\mathbf{U}_{ji}^T\right) \qquad [5.\ 13]$$

con i,k,m = 1,2,...,n

L-E 8. Obtener la matriz columna de fuerzas de Coriolis y centrípeta $\mathbf{H} = [h_i]^T$ cuyos elementos vienen definidos por:

$$h_i = \sum_{k=1}^{n}\sum_{m=1}^{n} h_{ikm}\dot{q}_k\dot{q}_m \qquad [5.\ 14]$$

L-E 9. Obtener la matriz columna de fuerzas de gravedad $\mathbf{C} = [c_i]^T$ cuyos elementos están definidos por:

$$c_i = \sum_{j=1}^{n}\left(-m_j\mathbf{g}\mathbf{U}_{ji}\,{}^j\mathbf{r}_j\right) \qquad [5.\ 15]$$

con i = 1,2,...,n

g: es el vector de gravedad expresado en el sistema de la base $\{S_0\}$y viene expresado por $(g_{x0}, g_{y0}, g_{z0}, 0)$

${}^i\mathbf{r}_j$: es el vector de coordenadas homogéneas del centro de masas del elemento j expresado en el sistema de referencia del elemento i.

L-E 10. La ecuación dinámica del sistema será:

$$\tau = \mathbf{D}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{C} \qquad [5.\ 16]$$

donde τ es el vector de fuerzas y pares motores efectivos aplicados sobre cada coordenada q_i.

NOTAS

1. La derivada de la matriz de D-H ${}^0\mathbf{A}_i$ respecto de la coordenada q_j puede obtenerse fácilmente de manera computacional, mediante la expresión:

$$\frac{\partial\, {}^0\mathbf{A}_i}{\partial q_j} = \begin{cases} {}^0\mathbf{A}_{j-1}\mathbf{Q}_j\,{}^{j-1}\mathbf{A}_i & \text{si } j \le i \\ [0] & \text{si } j > i \end{cases}$$

con:

$$\mathbf{Q}_i = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{si la articulación i es de rotación}$$

$$\mathbf{Q}_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{si la articulación i es de traslación}$$

2. Análogamente:

$$\frac{\partial \mathbf{U}_{ij}}{\partial q_k} = \frac{\partial}{\partial q_k}\left(\frac{\partial\, {}^0\mathbf{A}_i}{\partial q_j}\right) = \begin{cases} {}^0\mathbf{A}_{j-1}\mathbf{Q}_j\, {}^{j-1}\mathbf{A}_{k-1}\mathbf{Q}_k\, {}^{k-1}\mathbf{A}_i & \text{si } i \geq k \geq j \\ {}^0\mathbf{A}_{k-1}\mathbf{Q}_k\, {}^{k-1}\mathbf{A}_{j-1}\mathbf{Q}_j\, {}^{j-1}\mathbf{A}_i & \text{si } i \geq j \geq k \\ [0] & \text{si } k > i \text{ o } j > i \end{cases}$$

3. Las matrices $\mathbf{J}_i$ y $\mathbf{D}$ son simétricas y semidefinidas positivas.

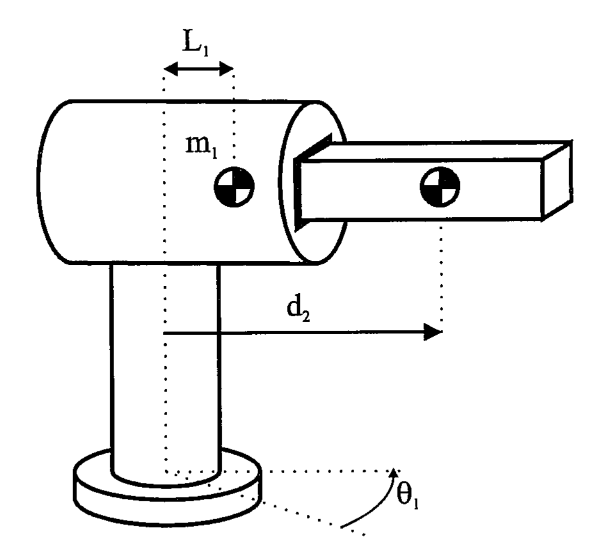


Figura 5.2. Robot polar de dos grados de libertad del Ejemplo 5.1.

4. El término h_{ikm} representa el efecto, en cuanto a fuerza o par, generado sobre el eslabón i como consecuencia del movimiento relativo entre los eslabones k y m. Se cumple que $h_{ikm}=h_{imk}$ y que $h_{iii}=0$.

5. En la obtención de las matrices de pseudoinercia $\mathbf{J}_i$, las integrales están extendidas al elemento i, de modo que ésta se evalúa para cada punto del elemento de masa dm y coordenadas (x_i y_i z_i) referidas al sistema de coordenadas del elemento.

Tabla 5.1. Parámetros D-H del robot polar del Ejemplo 5.1.

Articulación	θ_i	d_i	a_i	α_i
1	θ_1	0	0	-90
2	0	d_2	0	0

Ejemplo 5.1.

Se va a aplicar el método de Lagrange-Euler para la obtención del modelo dinámico del robot de 2 grados de libertad (θ_1, d_2) con base fija de la Figura 5.2.

L-E 1. Se asignan los sistemas de referencia y parámetros de Denavit-Hartenberg según la Figura 5.3 y la Tabla 5.1.

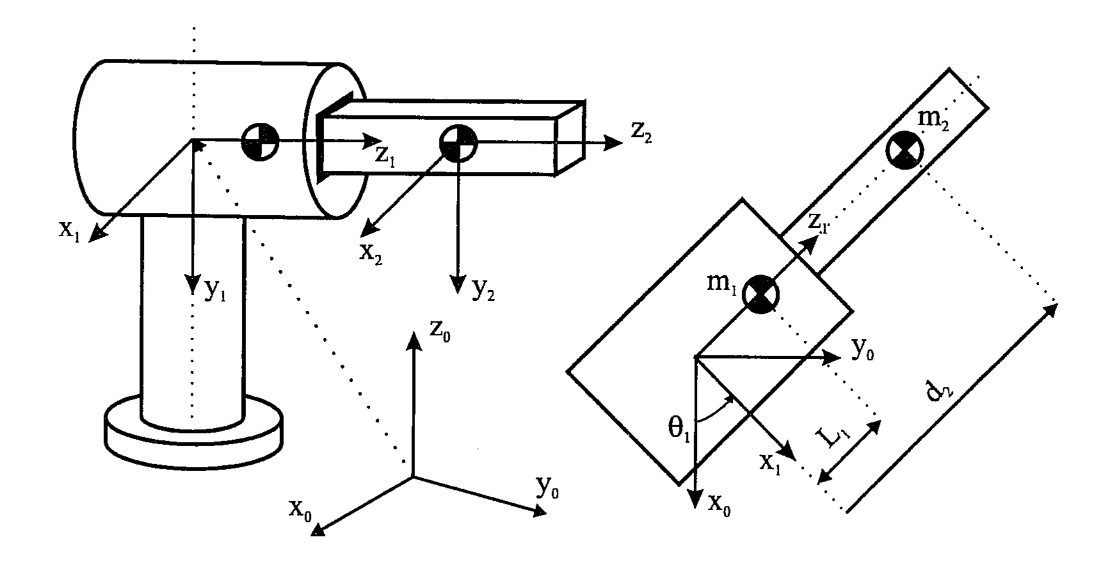


Figura 5.3. Sistemas de referencia del robot polar del Ejemplo 5.1.

L-E 2. Matrices de transformación ${}^{0}\mathbf{A}_i$

$$ {}^{0}\mathbf{A}_1 = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^{1}\mathbf{A}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

$$ {}^{0}\mathbf{A}_2 = {}^{0}\mathbf{A}_1 \, {}^{1}\mathbf{A}_2 = \begin{bmatrix} C_1 & 0 & -S_1 & -d_2S_1 \\ S_1 & 0 & C_1 & d_2C_1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

L-E 3. Matrices $\mathbf{U}_{ij}$

$$ \mathbf{U}_{11} = \frac{\partial\, {}^{0}\mathbf{A}_1}{\partial \theta_1} = \begin{bmatrix} -S_1 & 0 & -C_1 & 0 \\ C_1 & 0 & -S_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{U}_{12} = \frac{\partial\, {}^{0}\mathbf{A}_1}{\partial d_2} = [0] $$

$$ \mathbf{U}_{21} = \frac{\partial\, {}^{0}\mathbf{A}_2}{\partial \theta_1} = \begin{bmatrix} -S_1 & 0 & -C_1 & -d_2C_1 \\ C_1 & 0 & -S_1 & -d_2S_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{U}_{22} = \frac{\partial\, {}^{0}\mathbf{A}_2}{\partial d_2} = \begin{bmatrix} 0 & 0 & 0 & -S_1 \\ 0 & 0 & 0 & C_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} $$

L-E 4. Matrices $\mathbf{U}_{ijk}$

$$ \mathbf{U}_{111} = \frac{\partial \mathbf{U}_{11}}{\partial \theta_1} = \begin{bmatrix} -C_1 & 0 & S_1 & 0 \\ -S_1 & 0 & -C_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathbf{U}_{112} = \frac{\partial \mathbf{U}_{11}}{\partial d_2} = [0] $$

$$ \mathbf{U}_{121} = \frac{\partial \mathbf{U}_{12}}{\partial \theta_1} = [0] \qquad \mathbf{U}_{122} = \frac{\partial \mathbf{U}_{12}}{\partial d_2} = [0] $$

$$\mathbf{U}_{211}=\frac{\partial \mathbf{U}_{21}}{\partial \theta_1}=\begin{bmatrix}-C_1 & 0 & S_1 & d_2S_1\\ -S_1 & 0 & -C_1 & -d_2C_1\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\qquad \mathbf{U}_{212}=\frac{\partial \mathbf{U}_{21}}{\partial d_2}=\begin{bmatrix}0 & 0 & 0 & -C_1\\ 0 & 0 & 0 & -S_1\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}$$

$$\mathbf{U}_{221}=\frac{\partial \mathbf{U}_{22}}{\partial \theta_1}=\begin{bmatrix}0 & 0 & 0 & -C_1\\ 0 & 0 & 0 & -S_1\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix}\qquad \mathbf{U}_{222}=\frac{\partial \mathbf{U}_{22}}{\partial d_2}=[0]$$

L-E 5. Matrices de pseudoinercia $\mathbf{J}_i$

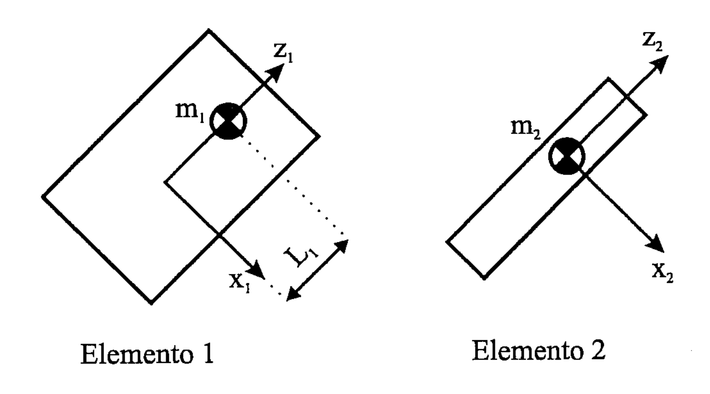


Figura 5.4. Elementos del robot polar del Ejemplo 5.1.

<u>Elemento 1</u>

$$\int_1 x_1^2 dm = 0 \qquad \int_1 y_1x_1 dm = \int_1 x_1y_1 dm = 0 \qquad \int_1 z_1x_1 dm = \int_1 x_1z_1 dm = 0$$

$$\int_1 y_1^2 dm = 0 \qquad \int_1 y_1z_1 dm = \int_1 z_1y_1 dm = 0$$

$$\int_1 z_1^2 dm = m_1L_1^2$$

$$\int_1 x_1 dm = 0 \qquad \int_1 y_1 dm = 0 \qquad \int_1 z_1 dm = L_1m_1 \qquad \int_1 dm = m_1$$

$$\mathbf{J}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & m_1L_1^2 & m_1L_1 \\ 0 & 0 & m_1L_1 & m_1 \end{bmatrix}$$

Elemento 2

Puesto que se considera la masa concentrada en el centro de masas y el origen del sistema de coordenadas del elemento 2 se toma en el mismo centro de masas, la matriz $\mathbf{J}_2$ toma la forma:

$$\mathbf{J}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix}$$

L-E 6. Matriz de inercias $\mathbf{D} = [d_{ij}]$

$$d_{11} = \sum_{k=\max(1,1)}^{2} \text{Traza}\left(\mathbf{U}_{k1}\mathbf{J}_k\mathbf{U}_{k1}^T\right) = \text{Tr}\left(\mathbf{U}_{11}\mathbf{J}_1\mathbf{U}_{11}^T\right) + \text{Tr}\left(\mathbf{U}_{21}\mathbf{J}_2\mathbf{U}_{21}^T\right) =$$

$$= \text{Tr}\begin{bmatrix} C_1^2L_1^2m_1 & S_1C_1L_1^2m_1 & 0 & 0 \\ C_1S_1L_1^2m_1 & S_1^2L_1^2m_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \text{Tr}\begin{bmatrix} C_1^2d_2^2m_2 & S_1C_1d_2^2m_2 & 0 & 0 \\ S_1C_1d_2^2m_2 & S_1^2d_2^2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} =$$

$$= \left(C_1^2 + S_1^2\right)m_1L_1^2 + \left(C_1^2 + S_1^2\right)d_2^2m_2 = m_1L_1^2 + m_2d_2^2$$

$$d_{12} = \sum_{k=\max(1,2)}^{2} \text{Traza}\left(\mathbf{U}_{k2}\mathbf{J}_k\mathbf{U}_{k1}^T\right) = \text{Tr}\left(\mathbf{U}_{22}\mathbf{J}_2\mathbf{U}_{21}^T\right) =$$

$$= \text{Tr}\begin{bmatrix} S_1C_1d_2m_2 & S_1^2d_2m_2 & 0 & 0 \\ -C_1^2d_2m_2 & -S_1C_1d_2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = S_1C_1d_2m_2 - S_1C_1d_2m_2 = 0$$

$$d_{21} = \sum_{k=\max(2,1)}^{2} \mathrm{Traza}\left(\mathbf{U}_{k1}\mathbf{J}_k\mathbf{U}_{k2}{}^{T}\right) = \mathrm{Tr}\left(\mathbf{U}_{21}\mathbf{J}_2\mathbf{U}_{22}{}^{T}\right) =$$

$$= \mathrm{Tr}\begin{bmatrix} S_1C_1d_2m_2 & -C_1{}^2d_2m_2 & 0 & 0 \\ S_1{}^2d_2m_2 & -S_1C_1d_2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = S_1C_1d_2m_2 - S_1C_1d_2m_2 = 0$$

$$d_{22} = \sum_{k=\max(2,2)}^{2} \mathrm{Traza}\left(\mathbf{U}_{k2}\mathbf{J}_k\mathbf{U}_{k2}{}^{T}\right) = \mathrm{Tr}\left(\mathbf{U}_{22}\mathbf{J}_2\mathbf{U}_{22}{}^{T}\right) =$$

$$= \mathrm{Tr}\begin{bmatrix} S_1{}^2m_2 & -S_1C_1m_2 & 0 & 0 \\ -S_1C_1m_2 & C_1{}^2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = S_1{}^2m_2 + C_1{}^2m_2 = m_2$$

luego:

$$\mathbf{D} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} = \begin{bmatrix} m_1L_1{}^2 + m_2d_2{}^2 & 0 \\ 0 & m_2 \end{bmatrix}$$

L-E 7. Términos en h_{ikm}

$$h_{111} = \sum_{j=\max(1,1,1)}^{2} \mathrm{Traza}\left(\mathbf{U}_{j11}\mathbf{J}_j\mathbf{U}_{j1}{}^{T}\right) = \mathrm{Tr}\left(\mathbf{U}_{111}\mathbf{J}_1\mathbf{U}_{11}{}^{T}\right) + \mathrm{Tr}\left(\mathbf{U}_{211}\mathbf{J}_2\mathbf{U}_{21}{}^{T}\right) =$$

$$= \mathrm{Tr}\begin{bmatrix} -C_1S_1m_1L_1{}^2 & -S_1{}^2m_1L_1{}^2 & 0 & 0 \\ C_1{}^2m_1L_1{}^2 & C_1S_1m_1L_1{}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \mathrm{Tr}\begin{bmatrix} -S_1Cd_2{}^2{}_1m_2 & -S_1{}^2d_2{}^2m_2 & 0 & 0 \\ C_1{}^2d_2{}^2m_2 & S_1C_1d_2{}^2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} =$$

$$= -C_1S_1m_1L_1{}^2 + C_1S_1m_1L_1{}^2 - d_2{}^2S_1C_1m_2 + d_2{}^2S_1C_1m_2 = 0$$

$$h_{112} = \sum_{j=\max(1,1,2)}^{2} \mathrm{Traza}\left(\mathbf{U}_{j12}\mathbf{J}_j\mathbf{U}_{j1}{}^{T}\right) = \mathrm{Tr}\left(\mathbf{U}_{212}\mathbf{J}_2\mathbf{U}_{21}{}^{T}\right) =$$

$$= \mathrm{Tr}\begin{bmatrix} C_1{}^2d_2m_2 & S_1C_1d_2m_2 & 0 & 0 \\ S_1C_1d_2m_2 & S_1{}^2d_2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = C_1{}^2d_2m_2 + S_1{}^2d_2m_2 = d_2m_2$$

$$h_{121} = \sum_{j=max(1,2,1)}^{2} \text{Traza}\left(\mathbf{U}_{j21}\mathbf{J}_j\mathbf{U}_{j1}^T\right) = \text{Tr}\left(\mathbf{U}_{221}\mathbf{J}_2\mathbf{U}_{21}^T\right)$$

como $\mathbf{U}_{221} = \mathbf{U}_{212} \quad \Rightarrow \quad h_{121} = h_{112} = d_2 m_2$

$$h_{211} = \sum_{j=max(2,1,1)}^{2} \text{Traza}\left(\mathbf{U}_{j11}\mathbf{J}_j\mathbf{U}_{j2}^T\right) = \text{Tr}\left(\mathbf{U}_{211}\mathbf{J}_2\mathbf{U}_{22}^T\right) =$$

$$= \text{Tr}\begin{bmatrix} -S_1^2 d_2 m_2 & S_1C_1d_2m_2 & 0 & 0 \\ S_1C_1d_2m_2 & -C_1^2d_2m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = S_1^2 d_2 m_2 - C_1^2 d_2 m_2 = -d_2 m_2$$

$$h_{122} = \sum_{j=max(1,2,2)}^{2} \text{Traza}\left(\mathbf{U}_{j22}\mathbf{J}_j\mathbf{U}_{j1}^T\right) = \text{Tr}\left(\mathbf{U}_{222}\mathbf{J}_2\mathbf{U}_{21}^T\right) = 0$$

$$h_{212} = \sum_{j=max(2,1,2)}^{2} \text{Traza}\left(\mathbf{U}_{j12}\mathbf{J}_j\mathbf{U}_{j2}^T\right) = \text{Tr}\left(\mathbf{U}_{212}\mathbf{J}_2\mathbf{U}_{22}^T\right) =$$

$$= \text{Tr}\begin{bmatrix} S_1C_1m_2 & -C_1^2m_2 & 0 & 0 \\ S_1^2m_2 & -S_1C_1m_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = S_1C_1m_2 - C_1S_1m_2 = 0$$

$$h_{221} = \sum_{j=max(2,2,1)}^{2} \text{Traza}\left(\mathbf{U}_{j21}\mathbf{J}_j\mathbf{U}_{j2}^T\right) = \text{Tr}\left(\mathbf{U}_{221}\mathbf{J}_2\mathbf{U}_{22}^T\right)$$

como $\mathbf{U}_{221} = \mathbf{U}_{212} \quad \Rightarrow \quad h_{221} = h_{212} = 0$

$$h_{222} = \sum_{j=max(2,2,2)}^{2} \text{Traza}\left(\mathbf{U}_{j22}\mathbf{J}_j\mathbf{U}_{j2}^T\right) = \text{Tr}\left(\mathbf{U}_{222}\mathbf{J}_2\mathbf{U}_{22}^T\right) = 0$$

L-E 8. Matriz columna de fuerzas de Coriolis y centrífugas $\mathbf{H} = [h_i]^T$

$$h_1 = \sum_{k=1}^{2}\sum_{m=1}^{2} h_{1km}\dot{q}_k\dot{q}_m = h_{111}\dot{\theta}_1\dot{\theta}_1 + h_{112}\dot{\theta}_1\dot{d}_2 + h_{121}\dot{d}_2\dot{\theta}_1 + h_{122}\dot{d}_2\dot{d}_2 =$$

$$= 0 \cdot \dot{\theta}_1^2 + (d_2m_2 + d_2m_2)\dot{\theta}_1\dot{d}_2 + 0 \cdot \dot{d}_2^2 = 2d_2m_2\dot{\theta}_1\dot{d}_2$$

$$h_2 = \sum_{k=1}^{2}\sum_{m=1}^{2} h_{2km}\,\dot{q}_k\dot{q}_m = h_{211}\dot{\theta}_1\dot{\theta}_1 + h_{212}\dot{\theta}_1\dot{d}_2 + h_{221}\dot{d}_2\dot{\theta}_1 + h_{222}\dot{d}_2\dot{d}_2 =$$

$$= -d_2 m_2 \dot{\theta}_1^{\,2} + (0+0)\dot{\theta}_1\dot{d}_2 + 0\cdot\dot{d}_2^{\,2} = -d_2 m_2 \dot{\theta}_1^{\,2}$$

luego

$$\mathbf{H} = \begin{bmatrix} 2d_2 m_2 \dot{\theta}_1 \dot{d}_2 \\ -d_2 m_2 \dot{\theta}_1^{\,2} \end{bmatrix}$$

L-E 9. Matriz columna de fuerzas de gravedad $\mathbf{C} = [c_i]^T$

$$c_i = \sum_{j=1}^{N}\left(-m_j \mathbf{g} \mathbf{U}_{ji}\,{}^{j}\mathbf{r}_j\right)$$

con :

g vector de gravedad expresado en el sistema de la base del robot $\{S_0\}$.

$$\mathbf{g} = [0,0,-g,0]$$

${}^{j}\mathbf{r}_j$ vector de coordenadas homogéneas de posición del centro de masas del eslabón j expresado en el sistema $\{S_j\}$.(Figura 5.4.)

$${}^{1}\mathbf{r}_1 = [0,0,L_1,1]^T$$
$${}^{2}\mathbf{r}_2 = [0,0,0,1]^T$$

luego:

$$c_1 = \sum_{j=1}^{2}\left(-m_j \mathbf{g} \mathbf{U}_{j1}\,{}^{j}\mathbf{r}_j\right) = -m_1 \mathbf{g} \mathbf{U}_{11}\,{}^{1}\mathbf{r}_1 - m_2 \mathbf{g} \mathbf{U}_{21}\,{}^{2}\mathbf{r}_2 =$$

$$= -m_1 \begin{bmatrix} 0 & 0 & -g & 0 \end{bmatrix} \begin{bmatrix} -S_1 & 0 & -C_1 & 0 \\ C_1 & 0 & -S_1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ L_1 \\ 1 \end{bmatrix} - m_2 \begin{bmatrix} 0 & 0 & -g & 0 \end{bmatrix} \begin{bmatrix} -S_1 & 0 & -C_1 & -d_2C_1 \\ C_1 & 0 & -S_1 & -d_2S_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

$$c_2 = \sum_{j=1}^{2}\left(-m_j \mathbf{g} \mathbf{U}_{j2}\,{}^{j}\mathbf{r}_j\right) = -m_1 \mathbf{g} \mathbf{U}_{12}\,{}^{1}\mathbf{r}_1 - m_2 \mathbf{g} \mathbf{U}_{22}\,{}^{2}\mathbf{r}_2 =$$

$$= -m_1 \begin{bmatrix} 0 & 0 & -g & 0 \end{bmatrix} [0] \begin{bmatrix} 0 \\ 0 \\ L_1 \\ 1 \end{bmatrix} - m_2 \begin{bmatrix} 0 & 0 & -g & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & -S_1 \\ 0 & 0 & 0 & C_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = 0$$

Por lo tanto:

$$\mathbf{C} = [c_i]^T = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

L-E 10. La ecuación dinámica del robot será :

$$\tau = \mathbf{D}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{C}$$

$$\begin{bmatrix} T_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} m_1 L_1^2 + m_2 d_2^2 & 0 \\ 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{d}_2 \end{bmatrix} + \begin{bmatrix} 2 d_2 m_2 \dot{\theta}_1 \dot{d}_2 \\ -d_2 m_2 \dot{\theta}_1^2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\begin{aligned} T_1 &= \left(m_1 L_1^2 + m_2 d_2^2\right)\ddot{\theta}_1 + 2 d_2 m_2 \dot{\theta}_1 \dot{d}_2 \\ F_2 &= m_2 \ddot{d}_2 - d_2 m_2 \dot{\theta}_1^2 \end{aligned} \qquad [5.\ 17]$$

Donde T_1 es el par motor efectivo (incluyendo rozamiento y otras perturbaciones) que actúa sobre la articulación 1 y F_2 es la fuerza motora efectiva que actúa sobre la articulación 2.

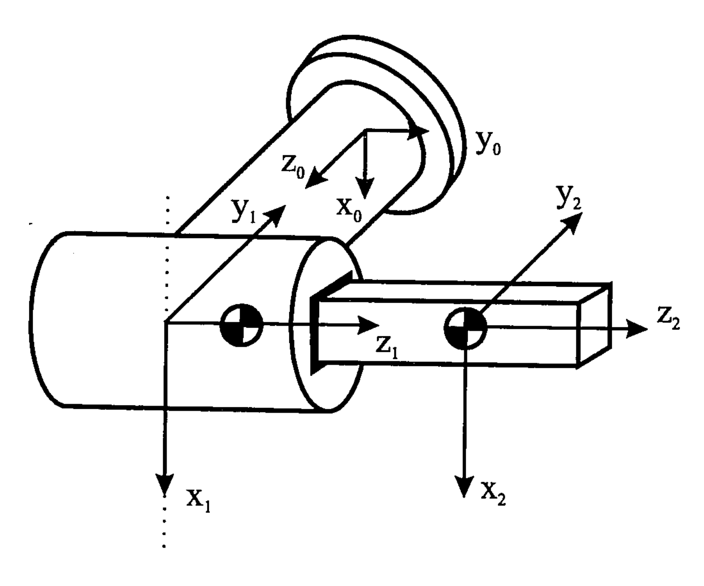


Figura 5.5. Robot polar del Ejemplo 5.1. en configuración horizontal.

Como se ve, el modelo dinámico responde a un sistema de 2 ecuaciones diferenciales de 2º orden no lineales y acopladas. Se observa también cómo el término gravedad **C** no aparece en el modelo. Esto es debido a que la estructura horizontal del robot permite que las fuerzas de gravedad se proyecten sobre los apoyos, no siendo necesario aportación de par o fuerza para vencerla. Si se considerase al robot en posición horizontal, tal y como aparece en Figura 5.5, manteniéndose la definición de los sistemas de referencia de la Figura 5.3, las expresiones de las matrices y vectores **A**, **U**, **J**, **D** y **H** resultarán iguales que en el caso anterior. Sin embargo, el vector **C** sí se vería afectado, pues ahora el vector de gravedad **g** expresado en el sistema de referencia de la base $\{S_0\}$ sería:

$$\mathbf{g} = [g, 0, 0, 0]$$

con lo que:

$$c_1 = -m_1 \begin{bmatrix} g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -C_1 L_1 \\ -S_1 L_1 \\ 0 \\ 0 \end{bmatrix} - m_2 \begin{bmatrix} g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -C_1 d_2 \\ -S_1 d_2 \\ 0 \\ 0 \end{bmatrix} = m_1 g L_1 C_1 + m_2 g d_2 C_1$$

$$c_2 = -m_1 \begin{bmatrix} g & 0 & 0 & 0 \end{bmatrix} [0] - m_2 \begin{bmatrix} g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} -S_1 \\ C_1 \\ 0 \\ 0 \end{bmatrix} = m_2 g S_1$$

$$\mathbf{C} = \begin{bmatrix} m_1 g L_1 C_1 + m_2 g d_2 C_1 \\ m_2 g S_1 \end{bmatrix}$$

De modo que las ecuaciones correspondientes al modelo dinámico tomarán ahora la siguiente forma:

$$\begin{aligned} T_1 &= \left(m_1 L_1^2 + m_2 d_2^2\right)\ddot{\theta}_1 + 2 d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 \\ F_2 &= m_2 \ddot{d}_2 - d_2 m_2 \dot{\theta}_1^2 + m_2 g \operatorname{sen}\theta_1 \end{aligned} \qquad [5.18]$$

5.3. OBTENCIÓN DEL MODELO DINÁMICO DE UN ROBOT MEDIANTE LA FORMULACIÓN DE NEWTON-EULER

La obtención del modelo dinámico de un robot a partir de la formulación Lagrangiana conduce a un algoritmo con un coste computacional de orden $O(n^4)$. Es decir, el número de operaciones a realizar crece con la potencia cuarta del número de grados de libertad. En el caso habitual de robots de 6 grados de libertad, este número de operaciones hace al algoritmo presentado en el epígrafe anterior materialmente inutilizable para ser utilizado en tiempo real.

La formulación de Newton-Euler parte del equilibrio de fuerzas y pares:

$$\sum \mathbf{F} = m\dot{v} \qquad \sum \mathbf{T} = \mathbf{I}\cdot\omega + \omega \times (\mathbf{I}\cdot\omega) \tag{5. 19}$$

Un adecuado desarrollo de estas ecuaciones conduce a una formulación recursiva en la que se obtienen la posición, velocidad y aceleración del eslabón i referidos a la base del robot a partir de los correspondientes del eslabón i-1 y del movimiento relativo de la articulación i. De este modo, partiendo del eslabón 1 se llega al eslabón n. Con estos datos se procede a obtener las fuerzas y pares actuantes sobre el eslabón i referidos a la base del robot a partir de los correspondientes al eslabón i+1, recorriéndose de esta forma todos los eslabones desde el eslabón n al eslabón 1.

El algoritmo se basa en operaciones vectoriales (con productos escalares y vectoriales entre magnitudes vectoriales, y productos de matrices con vectores) siendo más eficiente en comparación con las operaciones matriciales asociadas a la formulación Lagrangiana. De hecho, el orden de complejidad computacional de la formulación recursiva de Newton-Euler es O(n) lo que indica que depende directamente del número de grados de libertad.

El algoritmo se desarrolla en los siguientes pasos.

5.3.1. Algoritmo computacional para el modelado dinámico por Newton-Euler

N-E 1. Asignar a cada eslabón un sistema de referencia de acuerdo con las normas de D-H.

N-E 2. Obtener las matrices de rotación $^{i-1}\mathbf{R}_i$ y sus inversas $^{i}\mathbf{R}_{i-1} = \left(^{i-1}\mathbf{R}_i\right)^{-1} = \left(^{i-1}\mathbf{R}_i\right)^{T}$, siendo:

$$^{i-1}\mathbf{R}_i = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i \\ 0 & S\alpha_i & C\alpha_i \end{bmatrix}$$

N-E 3. Establecer las condiciones iniciales.

Para el sistema de la base $\{S_0\}$:

$^0\omega_0$: velocidad angular = $[0,0,0]^T$

$^0\dot{\omega}_0$: aceleracion angular = $[0,0,0]^T$

$^0\mathbf{v}_0$: velocidad lineal = $[0,0,0]^T$

$^0\dot{\mathbf{v}}_0$: aceleracion lineal = $[g_x, g_y, g_z]^T$

$^0\omega_0$, $^0\dot{\omega}_0$ y $^0\mathbf{v}_0$ son típicamente nulos salvo que la base del robot esté en movimiento.
Para el extremo del robot se conocerá la fuerza y el par ejercidos externamente $^{n+1}\mathbf{f}_{n+1}$ y $^{n+1}\mathbf{n}_{n+1}$

$\mathbf{z}_0 = [0,0,1]^T$

$^i\mathbf{p}_i$ = Coordenadas del origen del sistema $\{S_i\}$ respecto a $\{S_{i-1}\} = [a_i, d_i S_i, d_i C_i]$

$^i\mathbf{s}_i$ = Coordenadas del centro de masas del eslabon i respecto del sistema $\{S_i\}$

$^i\mathbf{I}_i$ = Matriz de inercia del eslabon i respecto del su centro de masas expresado en $\{S_i\}$

Para i = 1...n realizar los pasos 4 a 7 :

N-E 4. Obtener la velocidad angular del sistema $\{S_i\}$.

$$^i\omega_i = \begin{cases} ^i\mathbf{R}_{i-1}\left(^{i-1}\omega_{i-1} + \mathbf{z}_0\,\dot{q}_i\right) & \text{si el eslabón i es de rotación} \\ ^i\mathbf{R}_{i-1}\,^{i-1}\omega_{i-1} & \text{si el eslabón i es de traslación} \end{cases} \quad [5.20]$$

N-E 5. Obtener la aceleración angular del sistema $\{S_i\}$.

$$^i\dot{\omega}_i = \begin{cases} ^i\mathbf{R}_{i-1}\left(^{i-1}\dot{\omega}_{i-1} + \mathbf{z}_0\,\ddot{q}_i\right) + \,^{i-1}\omega_{i-1} \times \mathbf{z}_0\,\dot{q}_i & \text{si el eslabón i es de rotación} \\ ^i\mathbf{R}_{i-1}\,^{i-1}\dot{\omega}_{i-1} & \text{si el eslabón i es de traslación} \end{cases} \quad [5.21]$$

N-E 6. Obtener la aceleración lineal del sistema i:

$$
{}^{i}\dot{\mathbf{v}}_i = \begin{cases} {}^{i}\dot{\omega}_i \times {}^{i}\mathbf{p}_i + {}^{i}\omega_i \times \left({}^{i}\omega_i \times {}^{i}\mathbf{p}_i\right) + {}^{i}\mathbf{R}_{i-1}\, {}^{i-1}\dot{\mathbf{v}}_{i-1} & \text{si el eslabón i es de rotación} \\ {}^{i}\mathbf{R}_{i-1}\left(\mathbf{z}_0\, \ddot{q}_i + {}^{i-1}\dot{\mathbf{v}}_{i-1}\right) + {}^{i}\dot{\omega}_i \times {}^{i}\mathbf{p}_i + 2\, {}^{i}\omega_i \times {}^{i}\mathbf{R}_{i-1}\mathbf{z}_0\, \dot{q}_i + & \\ \quad + {}^{i}\omega_i \times \left({}^{i}\omega_i \times {}^{i}\mathbf{p}_i\right) & \text{si el eslabón i es de traslación} \end{cases} \qquad [5.22]
$$

N-E 7. Obtener la aceleración lineal del centro de gravedad del eslabón i:

$$
{}^{i}\mathbf{a}_i = {}^{i}\dot{\omega}_i \times {}^{i}\mathbf{S}_i + {}^{i}\omega_i \times \left({}^{i}\omega_i \times {}^{i}\mathbf{s}_i\right) + {}^{i}\dot{\mathbf{v}}_i \qquad [5.23]
$$

Para i = n...1 realizar los pasos 8 a10.

N-E 8. Obtener la fuerza ejercida sobre el eslabón i:

$$
{}^{i}\mathbf{f}_i = {}^{i}\mathbf{R}_{i+1}\, {}^{i+1}\mathbf{f}_{i+1} + m_i\, {}^{i}\mathbf{a}_i \qquad [5.24]
$$

N-E 9. Obtener el par ejercido sobre el eslabón i:

$$
{}^{i}\mathbf{n}_i = {}^{i}\mathbf{R}_{i+1}\left[{}^{i+1}\mathbf{n}_i + \left({}^{i+1}\mathbf{R}_i\, {}^{i}\mathbf{p}_i\right) \times {}^{i+1}\mathbf{f}_{i+1}\right] + \left({}^{i}\mathbf{p}_i + {}^{i}\mathbf{s}_i\right) \times m_i\, {}^{i}\mathbf{a}_i + {}^{i}\mathbf{I}_i\, {}^{i}\dot{\omega}_i + {}^{i}\omega_i \times \left({}^{i}\mathbf{I}_i\, {}^{i}\omega_i\right) \qquad [5.25]
$$

N-E 10. Obtener la fuerza o par aplicado a la articulación i

$$
\tau_i = \begin{cases} {}^{i}\mathbf{n}_i^T\; {}^{i}\mathbf{R}_{i-1}\mathbf{z}_0 & \text{si el eslabón i es de rotación} \\ {}^{i}\mathbf{f}_i^T\; {}^{i}\mathbf{R}_{i-1}\mathbf{z}_0 & \text{si el eslabón i es de traslación} \end{cases} \qquad [5.26]
$$

Donde τ es el par o fuerza efectivo (par motor menos pares de rozamiento o perturbación).

Tabla 5.2. Parámetros D-H del robot polar del Ejemplo 5.2.

Articulación	θ_i	d_i	a_i	α_i
1	θ_1	0	0	-90
2	0	d_2	0	0

Ejemplo 5.2.

En el epígrafe anterior se obtuvo el modelo dinámico del robot de la Figura 5.2 a través de la formulación de Lagrange-Euler. Se va a obtener aquí el modelo dinámico del mismo robot, utilizando la formulación de Newton-Euler.

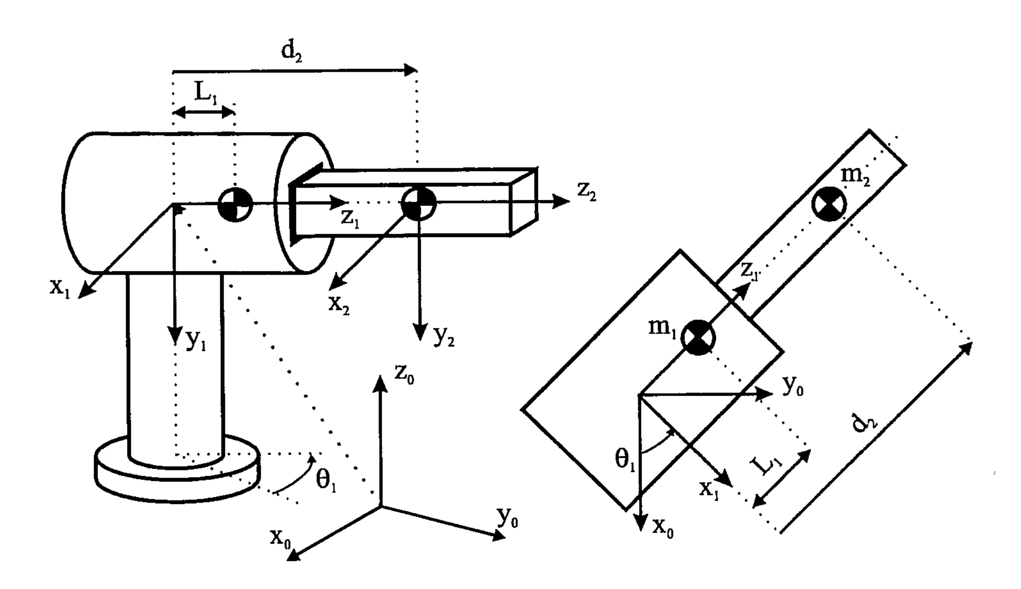


Figura 5.6. Configuración y ejes de referencia del robot polar del Ejemplo 5.2.

N-E 1. La asignación de los sistemas de referencia según D-H es la mostrada en la Figura 5.6. Los correspondientes parámetros de D-H se muestran en la Tabla 5.2.

N-E 2. Las matrices de rotación ${}^{i-1}\mathbf{R}_i$ y sus inversas son:

$$
{}^0\mathbf{R}_1 = \begin{bmatrix} C_1 & 0 & -S_1 \\ S_1 & 0 & C_1 \\ 0 & -1 & 0 \end{bmatrix} \qquad {}^1\mathbf{R}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad {}^0\mathbf{R}_2 = \begin{bmatrix} C_1 & 0 & -S_1 \\ S_1 & 0 & C_1 \\ 0 & -1 & 0 \end{bmatrix}
$$

$$
{}^1\mathbf{R}_0 = \begin{bmatrix} C_1 & S_1 & 0 \\ 0 & 0 & -1 \\ -S_1 & C_1 & 0 \end{bmatrix} \qquad {}^2\mathbf{R}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad {}^2\mathbf{R}_0 = \begin{bmatrix} C_1 & S_1 & 0 \\ 0 & 0 & -1 \\ -S_1 & C_1 & 0 \end{bmatrix}
$$

N-E 3.

$$
{}^0\omega_0 = [0,0,0]^T \qquad {}^0\dot{\omega}_0 = [0,0,0]^T
$$

$$
{}^0\mathbf{v}_0 = [0,0,0]^T \qquad {}^0\dot{\mathbf{v}}_0 = [0,0,g]^T
$$

y como no se ejercen fuerzas externas en el extremo del robot: ${}^3\mathbf{f}_3 = {}^3\mathbf{n}_3 = 0$

$$\mathbf{z}_0 = [0,0,1]^T$$

$$^1\mathbf{p}_1 = [0,0,0]^T \qquad {}^2\mathbf{p}_2 = [0,0,d_2]^T$$

$$^1\mathbf{S}_1 = [0,0,L_1] \qquad {}^2\mathbf{S}_2 = [0,0,0]$$

y por estar toda la masa de los elementos 1 y 2 concentrada en sus respectivos centros de gravedad:

$$^1\mathbf{I}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad {}^2\mathbf{I}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

N-E 4.

$$^1\omega_1 = {}^1\mathbf{R}_0\left({}^0\omega_0 + \mathbf{z}_0\dot{\theta}_1\right) = \begin{bmatrix} C_1 & S_1 & 0 \\ 0 & 0 & -1 \\ -S_1 & C_1 & 0 \end{bmatrix} \cdot \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ -\dot{\theta}_1 \\ 0 \end{bmatrix}$$

$$^2\omega_2 = {}^2\mathbf{R}_1{}^1\omega_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -\dot{\theta}_1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -\dot{\theta}_1 \\ 0 \end{bmatrix}$$

N-E 5.

$$^1\dot{\omega}_1 = {}^1\mathbf{R}_0\left({}^0\dot{\omega}_0 + \mathbf{z}_0\ddot{\theta}_1\right) + {}^0\omega_0 \times \mathbf{z}_0\dot{\theta}_1 =$$

$$= \begin{bmatrix} C_1 & S_1 & 0 \\ 0 & 0 & -1 \\ -S_1 & C_1 & 0 \end{bmatrix} \cdot \left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \ddot{\theta}_1 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\ddot{\theta}_1 \\ 0 \end{bmatrix}$$

$$^2\dot{\omega}_2 = {}^2\mathbf{R}_1{}^1\dot{\omega}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -\ddot{\theta}_1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -\ddot{\theta}_1 \\ 0 \end{bmatrix}$$

N-E 6.

$$^{1}\dot{\mathbf{v}}_1={}^{1}\dot{\boldsymbol{\omega}}_1\times{}^{1}\mathbf{p}_1+{}^{1}\boldsymbol{\omega}_1\times\left({}^{1}\boldsymbol{\omega}_1\times{}^{1}\mathbf{p}_1\right)+{}^{1}\mathbf{R}_0\,{}^{0}\dot{\mathbf{v}}_0=$$

$$=\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\0\end{bmatrix}+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\0\end{bmatrix}\right)+\begin{bmatrix}C_1&S_1&0\\0&0&-1\\-S_1&C_1&0\end{bmatrix}\begin{bmatrix}0\\0\\g\end{bmatrix}=\begin{bmatrix}0\\-g\\0\end{bmatrix}$$

$$^{2}\dot{\mathbf{v}}_2={}^{2}\mathbf{R}_1\left(\mathbf{z}_0\ddot{d}_2+{}^{1}\dot{\mathbf{v}}_1\right)+{}^{2}\dot{\boldsymbol{\omega}}_2\times{}^{2}\mathbf{p}_2+2\,{}^{2}\boldsymbol{\omega}_2\times\left({}^{2}\mathbf{R}_1\mathbf{z}_0\dot{d}_2\right)+{}^{2}\boldsymbol{\omega}_2\times\left({}^{2}\boldsymbol{\omega}_2\times{}^{2}\mathbf{p}_2\right)=$$

$$=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\left(\begin{bmatrix}0\\0\\\ddot{d}_2\end{bmatrix}+\begin{bmatrix}0\\-g\\0\end{bmatrix}\right)+\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\d_2\end{bmatrix}+2\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}0\\0\\\dot{d}_2\end{bmatrix}\right)+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\d_2\end{bmatrix}\right)=$$

$$=\begin{bmatrix}0\\-g\\\ddot{d}_2\end{bmatrix}+\begin{bmatrix}-\ddot{\theta}_1d_2\\0\\0\end{bmatrix}+2\begin{bmatrix}-\dot{\theta}_1\dot{d}_2\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0\\-\dot{\theta}_1^2d_2\end{bmatrix}=\begin{bmatrix}-\ddot{\theta}_1d_2-2\dot{\theta}_1\dot{d}_2\\-g\\\ddot{d}_2-\dot{\theta}_1^2d_2\end{bmatrix}$$

N-E 7.

$$^{1}\mathbf{a}_1={}^{1}\dot{\boldsymbol{\omega}}_1\times{}^{1}\mathbf{s}_1+{}^{1}\boldsymbol{\omega}_1\times\left({}^{1}\boldsymbol{\omega}_1\times{}^{1}\mathbf{s}_1\right)+{}^{1}\dot{\mathbf{v}}_1=$$

$$=\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\L_1\end{bmatrix}+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\L_1\end{bmatrix}\right)+\begin{bmatrix}0\\-g\\0\end{bmatrix}=\begin{bmatrix}-\ddot{\theta}_1L_1\\-g\\-\dot{\theta}_1{}^2L_1\end{bmatrix}$$

$$^{2}\mathbf{a}_2={}^{2}\dot{\boldsymbol{\omega}}_2\times{}^{2}\mathbf{s}_2+{}^{1}\boldsymbol{\omega}_2\times\left({}^{2}\boldsymbol{\omega}_2\times{}^{2}\mathbf{s}_2\right)+{}^{2}\dot{\mathbf{v}}_2=$$

$$=\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\0\end{bmatrix}+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\begin{bmatrix}0\\0\\0\end{bmatrix}\right)+\begin{bmatrix}-\ddot{\theta}_1d_2-2\dot{\theta}_1\dot{d}_2\\-g\\\ddot{d}_2-\dot{\theta}_1{}^2d_2\end{bmatrix}=\begin{bmatrix}-\ddot{\theta}_1d_2-2\dot{\theta}_1\dot{d}_2\\-g\\\ddot{d}_2-\dot{\theta}_1{}^2d_2\end{bmatrix}$$

N-E 8.

$$^{2}\mathbf{f}_2={}^{2}\mathbf{R}_3\,{}^{3}\mathbf{f}_3+m_2\,{}^{2}\mathbf{a}_2={}^{2}\mathbf{R}_3\begin{bmatrix}0\\0\\0\end{bmatrix}+m_2\begin{bmatrix}-\ddot{\theta}_1d_2-2\dot{\theta}_1\dot{d}_2\\-g\\\ddot{d}_2-\dot{\theta}_1{}^2d_2\end{bmatrix}=\begin{bmatrix}-\ddot{\theta}_1d_2m_2-2\dot{\theta}_1\dot{d}_2m_2\\-gm_2\\\ddot{d}_2m_2-\dot{\theta}_1{}^2d_2m_2\end{bmatrix}$$

$$
{}^{1}\mathbf{f}_1={}^{1}\mathbf{R}_2\,{}^{2}\mathbf{f}_2+m_1\,{}^{1}\mathbf{a}_1=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}-\ddot{\theta}_1 d_2 m_2-2\dot{\theta}_1\dot{d}_2 m_2\\-gm_2\\\ddot{d}_2 m_2-\dot{\theta}_1^{\,2} d_2 m_2\end{bmatrix}+m_1\begin{bmatrix}-\ddot{\theta}_1 l_1\\-g\\-\dot{\theta}_1^{\,2} l_1\end{bmatrix}=
$$

$$
=\begin{bmatrix}-\ddot{\theta}_1 d_2 m_2-2\dot{\theta}_1\dot{d}_2 m_2-\ddot{\theta}_1 l_1 m_1\\-g(m_1+m_2)\\\ddot{d}_2 m_2-\dot{\theta}_1^{\,2} d_2 m_2-\dot{\theta}_1^{\,2} l_1 m_1\end{bmatrix}
$$

N-E 9.

$$
{}^{2}\mathbf{n}_1={}^{2}\mathbf{R}_3\left[{}^{3}\mathbf{n}_3+\left({}^{3}\mathbf{R}_2\,{}^{2}\mathbf{p}_2\right)\times{}^{3}\mathbf{f}_3\right]+\left({}^{2}\mathbf{p}_2+{}^{2}\mathbf{s}_2\right)\times m_2\,{}^{2}\mathbf{a}_2+{}^{2}\mathbf{I}_2\,{}^{2}\dot{\omega}_2+{}^{2}\omega_2\times\left({}^{2}\mathbf{I}_2\cdot{}^{2}\omega_2\right)=
$$

$$
=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\cdot\left(\begin{bmatrix}0\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0\\0\end{bmatrix}\right)+\left(\begin{bmatrix}0\\0\\d_2\end{bmatrix}+\begin{bmatrix}0\\0\\0\end{bmatrix}\right)\times m_2\begin{bmatrix}-\ddot{\theta}_1 d_2-2\dot{\theta}_1\dot{d}_2\\-g\\\ddot{d}_2-\dot{\theta}_1^{\,2} d_2\end{bmatrix}+
$$

$$
+\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix}\cdot\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix}\cdot\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\right)=\begin{bmatrix}d_2 m_2 g\\\left(-\ddot{\theta}_1 d_2^{\,2}-2\dot{\theta}_1\dot{d}_2 d_2\right)m_2\\0\end{bmatrix}
$$

$$
{}^{1}\mathbf{n}_1={}^{1}\mathbf{R}_2\left[{}^{2}\mathbf{n}_2+\left({}^{2}\mathbf{R}_1\,{}^{1}\mathbf{p}_1\right)\times{}^{2}\mathbf{f}_2\right]+\left({}^{1}\mathbf{p}_1+{}^{1}\mathbf{s}_1\right)\times m_1\,{}^{1}\mathbf{a}_1+{}^{1}\mathbf{I}_1\,{}^{1}\dot{\omega}_1+{}^{1}\omega_1\times\left({}^{1}\mathbf{I}_1\,{}^{1}\omega_1\right)=
$$

$$
=\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\cdot\left(\begin{bmatrix}d_2 m_2 g\\\left(-\ddot{\theta}_1 d_2^{\,2}-2\dot{\theta}_1\dot{d}_2 d_2\right)m_2\\0\end{bmatrix}+\begin{bmatrix}0\\0\\0\end{bmatrix}\times\begin{bmatrix}-\ddot{\theta}_1 d_2 m_2-2\dot{\theta}_1\dot{d}_2 m_2\\-gm_2\\\ddot{d}_2 m_2-\dot{\theta}_1^{\,2} d_2 m_2\end{bmatrix}\right)+
$$

$$
+\left(\begin{bmatrix}0\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0\\L_1\end{bmatrix}\right)\times\begin{bmatrix}-\ddot{\theta}_1 L_1 m_1\\-gm_1\\-\dot{\theta}_1^2 L_1 m_1\end{bmatrix}+\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix}\cdot\begin{bmatrix}0\\-\ddot{\theta}_1\\0\end{bmatrix}+\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\times\left(\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix}\cdot\begin{bmatrix}0\\-\dot{\theta}_1\\0\end{bmatrix}\right)=
$$

$$
=\begin{bmatrix}d_2 m_2 g\\\left(-\ddot{\theta}_1 d_2^{\,2}-2\dot{\theta}_1\dot{d}_2 d_2\right)m_2\\0\end{bmatrix}+\begin{bmatrix}L_1 m_1 g\\-\ddot{\theta}_1 L_1^2 m_1\\0\end{bmatrix}=\begin{bmatrix}\left(d_2 m_2+L_1 m_1\right)g\\\left(-\ddot{\theta}_1 d_2^{\,2}-2\dot{\theta}_1\dot{d}_2 d_2\right)m_2-\ddot{\theta}_1 L_1^2 m_1\\0\end{bmatrix}
$$

N-E 10.

$$F_2 = {}^2\mathbf{f}_2^{T\,2}\mathbf{R}_1\mathbf{z}_0 = \begin{bmatrix} -\ddot{\theta}_1 d_2 m_2 - 2\dot{\theta}_1 \dot{d}_2 m_2 \\ -g m_2 \\ \ddot{d}_2 m_2 - \dot{\theta}_1^2 d_2 m_2 \end{bmatrix}^T \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \ddot{d}_2 m_2 - \dot{\theta}_1^2 d_2 m_2$$

$$T_1 = {}^1\mathbf{n}_1^{T\,1}\mathbf{R}_0\mathbf{z}_0 = \begin{bmatrix} (d_2 m_2 + L_1 m_1)g \\ (-\ddot{\theta}_1 d_2{}^2 - 2\dot{\theta}_1 \dot{d}_2 d_2)m_2 - \ddot{\theta}_1 L_1{}^2 m_1 \\ 0 \end{bmatrix}^T \cdot \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} =$$

$$= (\ddot{\theta}_1 d_2{}^2 + 2\dot{\theta}_1 \dot{d}_2 d_2)m_2 + \ddot{\theta}_1 L_1{}^2 m_1$$

Por lo tanto, las ecuaciones que componen el modelo dinámico son :

$$\begin{aligned} \mathbf{T}_1 &= (\ddot{\theta}_1 d_2{}^2 + 2\dot{\theta}_1 \dot{d}_2 d_2)m_2 + \ddot{\theta}_1 L_1{}^2 m_1 \\ \mathbf{F}_2 &= m_2 \ddot{d}_2 - m_2 d_2 \dot{\theta}_1{}^2 \end{aligned} \qquad [5.\ 27]$$

Ecuaciones que coinciden con las obtenidas [5.17] mediante el planteamiento de Lagrange-Euler.

5.4. MODELO DINÁMICO EN VARIABLES DE ESTADO

La ecuación [5.16] establece el modelo dinámico inverso de un robot, dando los pares y fuerzas que deben proporcionar los actuadores para que las variables articulares sigan una determinada trayectoria $\mathbf{q}(t)$:

$$\tau = \mathbf{D}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{C} \qquad [5.\ 28]$$

En esta expresión conviene recordar que la matriz de inercias **D** y la matriz columna de gravedad **C** dependen de los valores de **q**, y que la matriz columna de fuerzas de Coriolis y centrípetas **H** depende

de $\mathbf{q}$ y $\dot{\mathbf{q}}$. Asimismo, hay que tener presente que el vector de pares generalizados τ, presupone pares efectivos, por lo que de existir pares perturbadores o de rozamiento (viscoso o seco) éstos deberán ser tenidos en cuenta, siendo:

$$\tau = \tau_{\text{motor}} - \tau_{\text{perturbador}} - \tau_{\text{rozamiento viscoso}} - \tau_{\text{rozamiento seco}} \quad [5.29]$$

La expresión [5.28] es por tanto no lineal, no siendo trivial obtener a partir de ella el modelo dinámico directo que proporciona la trayectoria seguida como consecuencia de la aplicación de unos pares determinados τ. Para obtener este modelo directo, así como por su utilidad posterior en el desarrollo de alguna técnica concreta de control, puede ser conveniente obtener el modelo dinámico en variables de estado.

Las variables de estado naturales del sistema serán las posiciones y velocidades de cada una de las articulaciones, siendo por lo tanto el vector de estado $[\mathbf{q},\dot{\mathbf{q}}]^T$. La ecuación [5.28] puede expresarse como:

$$\mathbf{D}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{C} = \tau \quad \Rightarrow \quad \mathbf{D}\ddot{\mathbf{q}} + \mathbf{N} = \tau \quad \Rightarrow$$

$$\ddot{\mathbf{q}} = \mathbf{D}^{-1}[\tau - \mathbf{N}] \quad [5.30]$$

con $\mathbf{N} = \mathbf{H} + \mathbf{C}$. Y haciendo uso del vector de estado:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{q}\\ \dot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}\dot{\mathbf{q}}\\ -\mathbf{D}^{-1}\mathbf{N}\end{bmatrix} + \begin{bmatrix}0\\ \mathbf{D}^{-1}\end{bmatrix}\tau \quad [5.31]$$

donde $\mathbf{D}^{-1}$ es función de $\mathbf{q}$ y $\mathbf{N}$ es función de $\mathbf{q}$ y $\dot{\mathbf{q}}$. La expresión [5.31] puede ponerse también de la forma:

$$\begin{aligned} &\frac{d}{dt}\begin{bmatrix}\mathbf{q}\\ \dot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}0 & I\\ 0 & 0\end{bmatrix}\cdot\begin{bmatrix}\mathbf{q}\\ \dot{\mathbf{q}}\end{bmatrix} + \begin{bmatrix}0\\ I\end{bmatrix}\cdot\mathbf{u} \\ &\mathbf{u} = \mathbf{D}^{-1}(\tau - \mathbf{N}) \end{aligned} \quad [5.32]$$

Esta representación responde a una ecuación de estado lineal, habiéndose trasladado la no linealidad a la entrada $\mathbf{u}$. En la Figura 5.7 se representa el correspondiente diagrama de bloques, donde:

$$\mathbf{A} = \begin{bmatrix}0 & I\\ 0 & 0\end{bmatrix} \text{ matriz (2n x 2n)} \quad \text{y } \mathbf{B} = \begin{bmatrix}0\\ I\end{bmatrix} \text{ matriz (2n x n)}$$

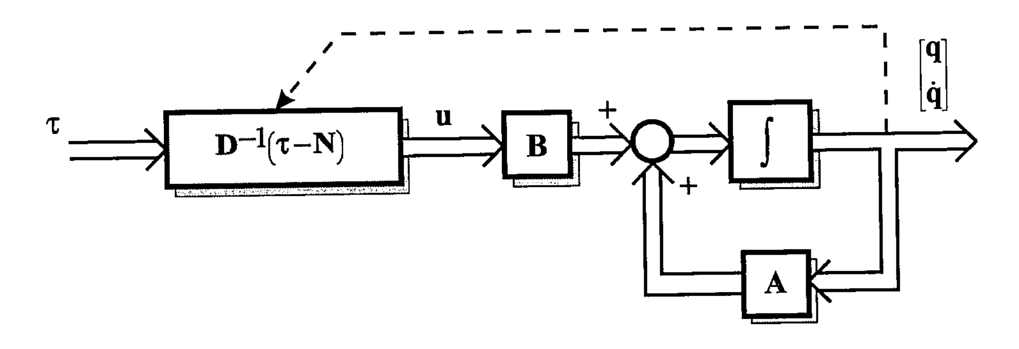


Figura 5.7. Diagrama de bloques de la dinámica de un robot en el espacio de estado.

El bloque sombreado es un bloque no lineal que depende del estado $[\mathbf{q},\dot{\mathbf{q}}]$, lo que se representa mediante línea discontinua.

Ejemplo 5.3.

Para el robot del Ejemplo 5.1 en su disposición horizontal, se obtuvo el modelo dinámico inverso dado por la ecuaciones [5.18]:

$$\begin{aligned} T_1 &= \left(m_1 L_1^2 + m_2 d_2^2\right)\ddot{\theta}_1 + 2d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 \\ F_2 &= m_2 \ddot{d}_2 - d_2 m_2 \dot{\theta}_1^2 + m_2 g \,\mathrm{sen}\,\theta_1 \end{aligned} \qquad [5.33]$$

o expresadas en forma matricial:

$$\begin{bmatrix} T_1 \\ F_2 \end{bmatrix} = \mathbf{D} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{d}_2 \end{bmatrix} + \mathbf{N}$$

con $\mathbf{D} = \begin{bmatrix} m_1 L_1^2 + m_2 d_2^2 & 0 \\ 0 & m_2 \end{bmatrix}$

$$\mathbf{N} = \begin{bmatrix} 2d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 \\ - d_2 m_2 \dot{\theta}_1^2 + m_2 g \,\mathrm{sen}\,\theta_1 \end{bmatrix}$$

Se tomará como vector de estado $\left[\theta_1, d_2, \dot{\theta}_1, \dot{d}_2\right]^T$, resultando:

$$\mathbf{D}^{-1} = \begin{bmatrix} \dfrac{1}{m_1 L_1^{\,2} + m_2 d_2^{\,2}} & 0 \\ 0 & \dfrac{1}{m_2} \end{bmatrix}$$

$$\mathbf{D}^{-1}(\mathbf{N} - \tau) = \begin{bmatrix} \dfrac{1}{m_1 L_1^{\,2} + m_2 d_2^{\,2}} & 0 \\ 0 & \dfrac{1}{m_2} \end{bmatrix} \begin{bmatrix} 2d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 - T_1 \\ -d_2 m_2 \dot{\theta}_1^{\,2} + m_2 g \,\mathrm{sen}\,\theta_1 - F_2 \end{bmatrix} =$$

$$= \begin{bmatrix} \dfrac{2d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 - T_1}{m_1 L_1^{\,2} + m_2 d_2^{\,2}} \\ \dfrac{-d_2 m_2 \dot{\theta}_1^{\,2} + m_2 g \,\mathrm{sen}\,\theta_1 - F_2}{m_2} \end{bmatrix}$$

La ecuación de estado tomará la forma:

$$\begin{bmatrix} \dot{\theta}_1 \\ \dot{d}_2 \\ \ddot{\theta}_1 \\ \ddot{d}_2 \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} \theta_1 \\ d_2 \\ \dot{\theta}_1 \\ \dot{d}_2 \end{bmatrix} + \mathbf{B} \cdot \mathbf{u}$$

$$\mathbf{u} = \begin{bmatrix} -\dfrac{2d_2 m_2 \dot{\theta}_1 \dot{d}_2 + m_1 g L_1 \cos\theta_1 + m_2 g d_2 \cos\theta_1 - \mathbf{T}_1}{m_1 L_1^{\,2} + m_2 d_2^{\,2}} \\ -\dfrac{-d_2 m_2 \dot{\theta}_1^{\,2} + m_2 g \,\mathrm{sen}\,\theta_1 - \mathbf{F}_2}{m_2} \end{bmatrix} \qquad [5.\ 34]$$

con:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{y} \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Esta expresión permitiría resolver computacionalmente el modelo dinámico directo. Partiendo de unas condiciones iniciales de posición $[\theta_1, d_2]^T$ y velocidad $[\dot{\theta}_1, \dot{d}_2]^T$ y de una secuencia de pares y fuerzas (T_1, F_2) a lo largo del tiempo, se evaluaría para el instante inicial el vector **u** y a partir de él, mediante [5.34], el vector derivada del estado $[\dot{\theta}_1, \dot{d}_2, \ddot{\theta}_1, \ddot{d}_2]$.

Puesto que:

$$\theta_1(k) = \dot{\theta}_1(k-1)\Delta t + \theta_1(k-1)$$
$$d_2(k) = \dot{d}_2(k-1)\Delta t + \theta_2(k-1)$$

podrían evaluarse los valores de la posición $[\theta_1, d_2]^T$ en un instante Δt posterior, y conocidos éstos y los nuevos valores de las velocidades $[\dot{\theta}_1, \dot{d}_2]^T$ se repetiría el proceso para el instante siguiente.

5.5. MODELO DINÁMICO EN EL ESPACIO DE LA TAREA

El modelo dinámico obtenido relaciona coordenadas articulares con pares o fuerzas desarrolladas en las articulaciones. En ocasiones es conveniente tener el modelo dinámico expresado como una relación entre la trayectoria del extremo del robot y las fuerzas y pares que en él se aplican, referidos todos a un sistema de coordenadas cartesianas fijo del entorno de trabajo. Cuando los datos (trayectorias, fuerzas, pares, etc.) se dan en estas coordenadas, se dice que se trabaja en el espacio de la tarea.

Para obtener esta expresión del modelo dinámico se partirá de la expresión [5.16] y de la relación [4.60] que establece las relación entre velocidades cartesianas y articulares haciendo uso de la matriz Jacobiana. Según ésta:

$$\dot{\mathbf{j}} = \mathbf{J}\,\dot{\mathbf{q}} \qquad [5.35]$$

donde $\mathbf{j}$ representa el vector de velocidades cartesianas del extremo del robot referidas al sistema de coordenadas asociado a su base $\dot{\mathbf{j}} = [\dot{x}, \dot{y}, \dot{z}, \dot{\alpha}, \dot{\beta}, \dot{\gamma}]^T$. Derivando esta expresión se obtiene:

$$\ddot{\mathbf{j}} = \dot{\mathbf{J}}\,\dot{\mathbf{q}} + \mathbf{J}\,\ddot{\mathbf{q}} \quad \Rightarrow \quad \ddot{\mathbf{q}} = \mathbf{J}^{-1}\,\ddot{\mathbf{j}} - \mathbf{J}^{-1}\dot{\mathbf{J}}\,\dot{\mathbf{q}} \qquad [5.36]$$

Expresiones que relacionan las aceleraciones cartesianas y articulares de manera directa e inversa. Por otra parte, partiendo del hecho de que la potencia consumida por el robot debe ser la misma tanto si se evalúa en el espacio cartesiano como en el articular, se tiene :

$$\text{Potencia} = \text{Par} \cdot \text{velocidad} \quad \Rightarrow \quad \mathbf{T}^T\,\dot{\mathbf{j}} = \tau^T\,\dot{\mathbf{q}} \qquad [5.37]$$

donde $\mathbf{T}^T$ es el vector de fuerzas y pares ejercidos en el extremo del robot expresado en el sistema de coordenadas de la base y τ^T el vector de fuerzas y pares ejercidas en las articulaciones. De [5.35] y [5.37] se obtiene:

$$\mathbf{T}^T\dot{\mathbf{j}} = \tau^T\dot{\mathbf{q}} \;\Rightarrow\; \mathbf{T}^T\mathbf{J}\dot{\mathbf{q}} = \tau^T\dot{\mathbf{q}} \;\Rightarrow\; \mathbf{T}^T\mathbf{J} = \tau^T \;\Rightarrow\; \tau = \mathbf{J}^T\mathbf{T} \qquad [5.38]$$

Expresión, de destacable utilidad, que relaciona los pares generalizados ejercidos en el extremo del robot con los ejercidos en cada una de las articulaciones.

Sustituyendo [5.36] y [5.38] en la ecuación del modelo dinámico [5.16] se obtiene:

$$\begin{aligned}
&\tau = \mathbf{D}\ddot{\mathbf{q}} + \mathbf{H} + \mathbf{C} \;\Rightarrow\\
&\mathbf{J}^T\mathbf{T} = c\mathbf{J}^{-1}\ddot{\mathbf{j}} - \mathbf{D}\mathbf{J}^{-1}\dot{\mathbf{J}}\dot{\mathbf{q}} + \mathbf{H} + \mathbf{C} \;\Rightarrow\\
&\mathbf{T} = \left(\mathbf{J}^T\right)^{-1}\mathbf{D}\mathbf{J}^{-1}\ddot{\mathbf{j}} - \left(\mathbf{J}^T\right)^{-1}\mathbf{D}\mathbf{J}^{-1}\dot{\mathbf{J}}\dot{\mathbf{q}} + \left(\mathbf{J}^T\right)^{-1}\mathbf{H} + \left(\mathbf{J}^T\right)^{-1}\mathbf{C} \;\Rightarrow
\end{aligned}$$

$$\mathbf{T} = \mathbf{D}_j\ddot{\mathbf{j}} + \mathbf{H}_j + \mathbf{C}_j \qquad [5.39]$$

con

$$\begin{cases}
\mathbf{D}_j = \left(\mathbf{J}^T\right)^{-1}\mathbf{D}\mathbf{J}^{-1}\\
\mathbf{H}_j = \left(\mathbf{J}^T\right)^{-1}\left(\mathbf{H} - \mathbf{D}\mathbf{J}^{-1}\dot{\mathbf{J}}\dot{\mathbf{q}}\right)\\
\mathbf{C}_j = \left(\mathbf{J}^T\right)^{-1}\mathbf{C}
\end{cases} \qquad [5.40]$$

Las expresiones [5.39] y [5.40] definen el modelo dinámico en coordenadas cartesianas o de la tarea de un robot a partir de su modelo dinámico en el espacio articular o de la configuración y de su matriz Jacobiana.

5.6. MODELO DINÁMICO DE LOS ACTUADORES

Como se ha indicado en la introducción de este capítulo, el modelo dinámico de un robot se compone por una parte del modelo de su estructura mecánica, que relaciona su movimiento con las fuerzas y pares que lo originan, y por otra parte del modelo de su sistema de accionamiento, que relaciona las órdenes de mando generadas en la unidad de control con las fuerzas y pares utilizados para producir el movimiento.

En el Capítulo 2, dedicado a la morfología del robot, se indicó que son los actuadores eléctricos de corriente continua los más utilizados en la actualidad, si bien es notable la tendencia a sustituir éstos por motores sin escobillas. En un caso u otro, el modelo dinámico del actuador responde a ecuaciones similares, por lo que a efectos de establecerlo se considerará el de motor de corriente continua.

Por su parte, los actuadores hidráulicos son usados en robots en los que la relación peso manipulable/peso del robot deba ser elevada. El modelo dinámico de un actuador hidráulico es significativamente más complejo que el de un actuador eléctrico. A las características dinámicas del conjunto ser-

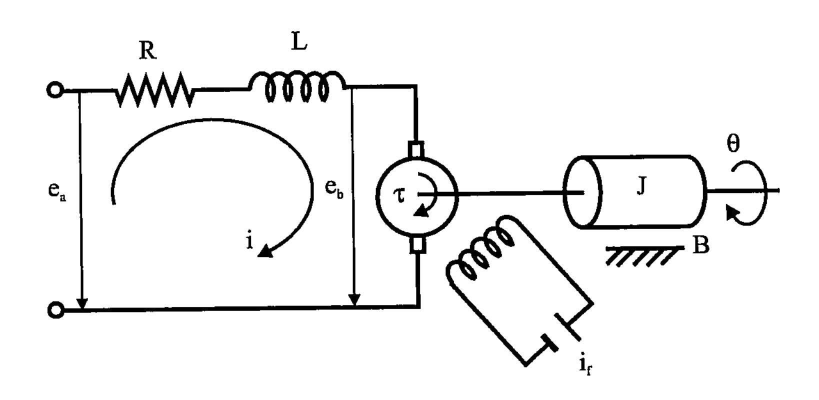


Figura 5.8. Esquema de funcionamiento de un motor de corriente continua controlado por inducido.

vo-válvula cilindro (o motor) se le debe incorporar el comportamiento no invariante del fluido (aceite), cuyas constantes dinámicas (índice de Bulk, viscosidad, etc.) varían notablemente con la temperatura. Por último, las propias líneas de transmisión, tuberías o mangueras, que canalizan al fluido desde la bomba a las servo-válvulas y de éstas a los actuadores, pueden influir en el comportamiento dinámico del conjunto.

Se van a exponer a continuación los modelos dinámicos que son admitidos para trabajar con ambos tipos de actuadores, debiendo quedar claro que en ambos casos se han realizado cierto número de simplificaciones que, por lo general, son válidas.

5.6.1. Motor eléctrico de corriente continua

Un accionamiento eléctrico de corriente continua consta de un motor de corriente continua alimentado por una etapa de potencia y controlado por un dispositivo analógico o digital. El modelado del motor de corriente continua controlado por inducido (Figura 5.8) responde a las ecuaciones que a continuación se deducen.

Cuando el rotor gira, se induce en él una tensión e_b directamente proporcional a la velocidad angular y que se conoce como fuerza contraelectromotriz :

$$e_b = k_b\dot{\theta} \qquad [5.41]$$

La velocidad de giro se controla mediante la tensión e_a, salida del amplificador de potencia. La ecuación diferencial del circuito del rotor es :

$$L_a\dot{i} + Ri + e_b = e_a \qquad [5.42]$$

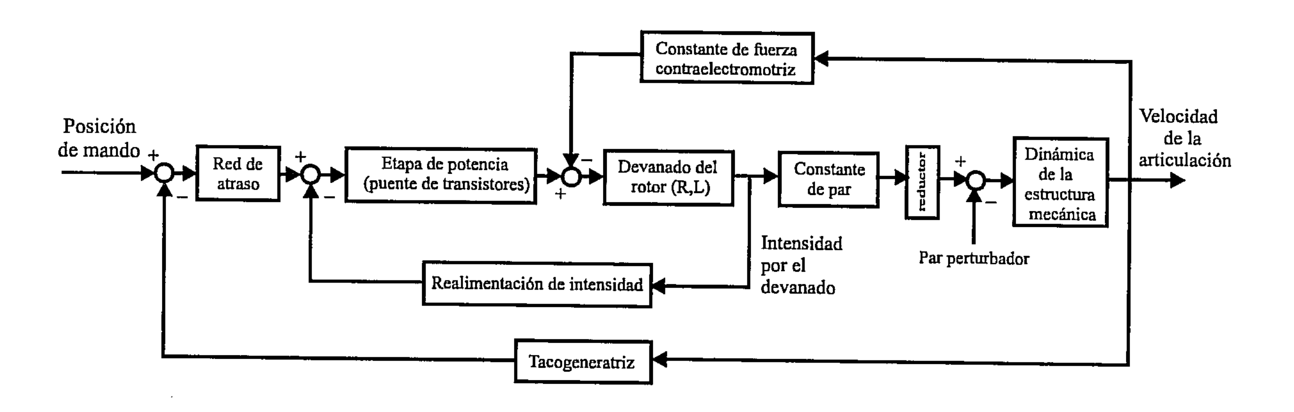


Figura 5.9. Esquema de bloques del modelo de motor eléctrico de corriente contínua.

Por otra parte, el motor desarrolla un par proporcional al producto del flujo en el entrehierro ψ y la intensidad i, siendo el flujo en el entrehierro:

$$\psi = k_f i_f \qquad [5.43]$$

donde i_f es la corriente de campo. De esta manera, la expresión del par desarrollado por el motor es la siguiente :

$$\tau = k_1 i \psi \qquad [5.44]$$

Para una corriente de campo i_f constante, el flujo se vuelve constante, y el par es directamente proporcional a la corriente que circula por el rotor:

$$\tau = k_p i \qquad [5.45]$$

Este par se emplea en vencer la inercia y la fricción, además de posibles pares perturbadores:

$$J\ddot{\theta} + B\dot{\theta} = \tau - \tau_p \qquad [5.46]$$

Por lo tanto, las ecuaciones del motor de corriente continua controlado por inducción son:

$$
\begin{aligned}
e_b &= k_b\dot{\theta} \\
(Ls+R)i + e_b &= e_a \\
\tau &= k_p i \\
\dot{\theta} &= \frac{\tau - \tau_p}{Js+B}
\end{aligned}
\qquad [5.47]
$$

donde todas las variables son en transformada de Laplace.

Para el control del motor se incluyen las etapas de potencia y control, utilizándose realimentación de intensidad y velocidad, tal y como se representa en la Figura 5.9.

En la Figura 5.10 se ha representado el diagrama de bloques correspondiente haciendo uso de funciones de transferencia [OGATA-93], [D'AZZO-95] y [OGATA-92], donde pueden realizarse ciertas simplificaciones:

$$
\begin{aligned}
&G_1 = k\frac{s+a}{s+b} \approx k_1 \\
&G_2 = k_2 \\
&L \approx 0
\end{aligned}
$$

J, B: Inercia y rozamiento viscoso equivalentes vistos a la salida del eje del rotor

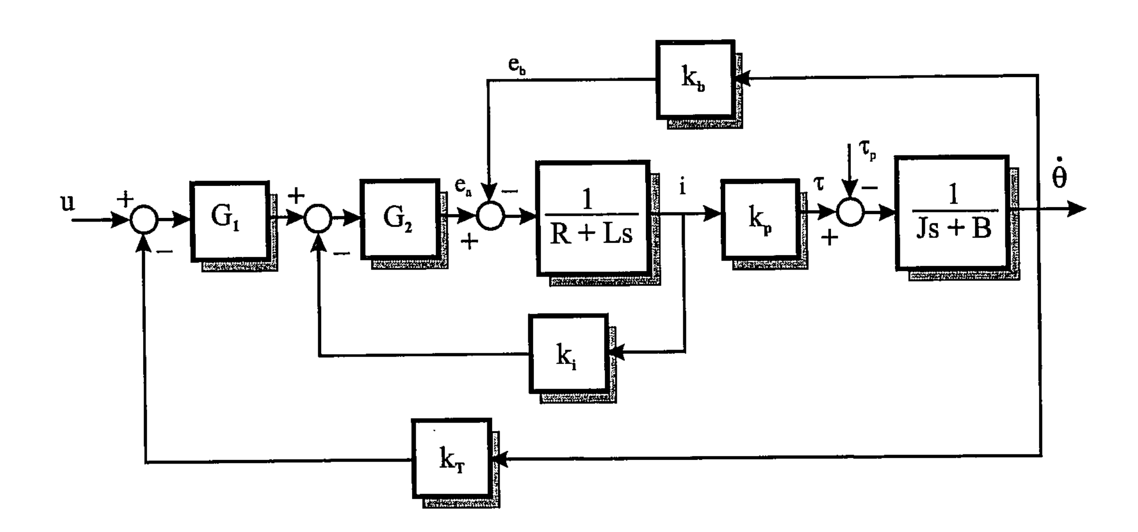


Figura 5.10. Diagrama de bloques del modelo de motor eléctrico de corriente continua.

Las simplificaciones del anterior diagrama permiten obtener (en ausencia de perturbaciones):

$$\frac{\dot{\theta}(s)}{u(s)} = \frac{k_p k_I k_2}{(R + k_i k_2)(Js + B) + k_p(k_b + k_T k_I k_2)} = \frac{k_m}{T_m s + 1} \qquad [5.48]$$

$$\frac{T(s)}{u(s)} = \frac{k_p k_I k_2 (Js + B)}{(R + k_i k_2)(Js + B) + k_p(k_b + k_T k_I k_2)} = \hat{k}_m \frac{\hat{T}_m s + 1}{T_m s + 1} \qquad [5.49]$$

donde se tiene que:

$$k_m = \frac{k_p k_I k_2}{k_p(k_b + k_T k_I k_2) + B(R + k_i k_2)}$$
$$\hat{k}_m = \frac{B k_p k_I k_2}{k_p(k_b + k_T k_I k_2) + B(R + k_i k_2)} \qquad [5.50]$$

$$T_m = \frac{J(R + k_i k_2)}{k_p(k_b + k_T k_I k_2) + B(R + k_i k_2)}$$
$$\hat{T}_m = \frac{J}{B} \qquad [5.51]$$

Se observa, por lo tanto, que el comportamiento tension-velocidad del motor de corriente continua responde al de un sistema de primer orden. En cuanto a la relación tensión-par, responde a un par polo-cero. En la práctica, la calidad de los motores utilizados en servoaccionamientos y las elevadas prestaciones de sus sistemas de control, hace que esta relación pueda considerarse casi constante (sin la dinámica propia de los polos y ceros).

5.6.2. Motor hidráulico con servoválvula

La introducción de sistemas electrónicos analógicos, y recientemente digitales, para el control de las válvulas de distribución de caudal utilizadas en los accionamiento hidráulicos (lineales o rotativos), ha permitido la evolución de las válvulas proporcionales a las servoválvulas, consiguiendo que el comportamiento dinámico de los actuadores hidráulicos tenga la calidad adecuada para ser usados en servomecanismos, y en especial en robótica.

El conjunto equipo electrónico, servoválvula y motor hidráulico (Figura 5.11) puede ser modelado en una versión simplificada según las siguientes ecuaciones:

Equilibrio de pares:

$$\tau = J\ddot{\theta} + B\dot{\theta} + \tau_p \quad [5.52]$$

Par desarrollado por el motor:

$$\tau = k_p \Delta P \quad [5.53]$$

Continuidad de caudales:

$$Q_1 = \dot{v}_1 + Q_f + Q_c \quad [5.54]$$

Caudal de fuga:

$$Q_f = k_f \Delta P \quad [5.55]$$

Pérdida de caudal por compresión del fluido:

$$Q_c = k_c \Delta\dot{P} \quad [5.56]$$

Electrónica de mando de la corredera de la servoválvula:

$$y = g(u) \quad [5.57]$$

Caudal suministrado por la servoválvula:

$$Q_1 = f(y) \cdot \sqrt{\Delta P} \quad [5.58]$$

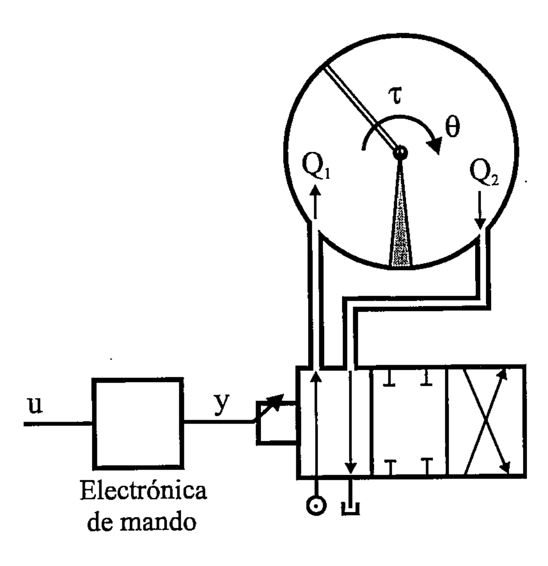


Figura 5.11. Esquema de un actuador hidráulico (motor hidráulico).

donde:

θ: ángulo girado por la paleta (y el eje) del rotor.
τ: par proporcionado por el motor.
τ_p: par externo perturbador.
J, B: inercia y constante de rozamiento viscoso (de motor y carga) asociados a la articulación.
ΔP: diferencia de presión entre las dos cámaras del motor.
Q_l: caudal proporcionado por la servoválvula (entrada al motor).
Q_f: caudal que se fuga entre las dos cámaras del motor (leakage).
Q_c: caudal perdido por la compresibilidad del fluido (depende del índice de Bulk, del volumen y de la diferencia de presiones).
v_l: volumen en la cámara de entrada al motor.
y: posición de la corredera de la servoválvula.
u: tensión de referencia a la electrónica de mando de la servovávula.
k_p, k_f y k_c se considerarán constantes.

El dispositivo electrónico de mando de la corredera de la válvula se diseña con el objetivo de que la relación entre señal de mando 'u' y posición de la corredera 'y' sea lo más parecida posible a una constante. Si bien este objetivo no es del todo alcanzable, existiendo siempre una cierta dinámica en dicha relación, la velocidad de la misma es muy superior a la dinámica propia del accionamiento hidráulico y de la articulación, por lo que puede ser considerada como constante.

Por este motivo, la relación [5.57] se sustituye por:

$$y = k'_1 \cdot u \qquad [5.59]$$

Por otra parte, la relación entre la posición de la corredera de la válvula, la diferencia de presiones y el caudal suministrado [5.58] es en principio no lineal. Linealizando entorno a un punto de funcionamiento se obtiene:

$$Q_l = k'_2\, y - k_i \Delta P \qquad [5.60]$$

Por lo tanto, la relación entre la señal de mando 'u', el caudal Q_l y la presión diferencial ΔP será:

$$Q_l = k_1 u - k_i \Delta P \qquad [5.61]$$

Por último, la velocidad de variación de volumen en la cámara del motor $\dot{v}_l$ será proporcional a la velocidad de giro de la paleta, luego:

$$\dot{v}_l = k_b \dot{\theta} \qquad [5.62]$$

Con lo que las ecuaciones [5.54], [5.55] y [5.56] podrán agruparse como:

$$Q_1 = k_b\dot{\theta} + k_f\Delta P + k_c\Delta\dot{P} \qquad [5.63]$$

Transformando por Laplace las ecuaciones [5.52], [5.53], [5.61] y [5.63] se obtiene el diagrama de bloques de la Figura 5.12, que como se observa presenta una absoluta analogía con el correspondiente a un accionamiento eléctrico (Figura 5.10) salvo por la realimentación de velocidad de giro del actuador presente en aquél. Esta última puede ser incluida en la electrónica de mando, siendo entonces el modelado de ambos actuadores equivalente aunque con características dinámicas y posibilidades diferentes.

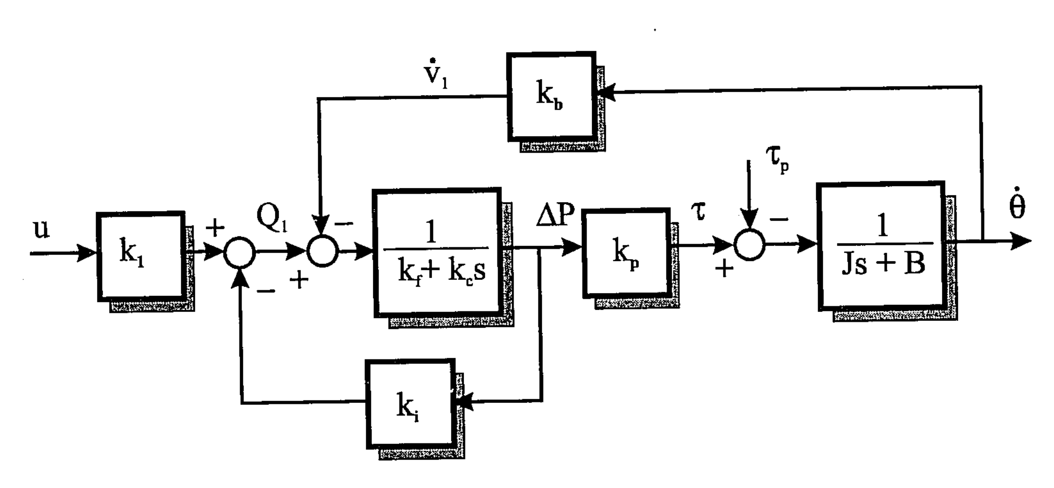


Figura 5.12. Diagrama de bloques del modelo de actuador hidráulico.

En el caso de utilizar un cilindro hidráulico el modelado se hace más complicado, motivado entre otras razones por la diferencia de áreas del émbolo en ambas cámaras, lo que hace que su funcionamiento a extensión y retracción sea notablemente diferente.

Estudios más detallados del modelado de accionamientos hidráulicos pueden encontrarse en [WATTON-89].

5.7. BIBLIOGRAFÍA

[AMARIS-95] H. Amaris, *Modelado y control de la flexión de robots de grandes dimensiones*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1995.

[CRAIG-89] J.J. Craig, “Manipulator Dynamics”, en: *Introduction to Robotics*, Addsison Wesley, 2ª edición, 1989.

[D'AZZO-95] J.J. D'Azzo y C.H. Houpis, "Writing System Equations", en: *Linear Control System Analysis and Design*, McGraw-Hill, 4ª edición, 1995

[FU-88] K.S. Fu, R.C. González y C.S.G. Lee, "Dinámica del Brazo del Robot", en: *Robótica, Control, Detección, Visión e Inteligencia*, McGraw-Hill/Interamericana, Madrid, 1988

[OGATA-92] K. Ogata, "Electromechanical Systems", en: *System Dynamics*, Prentice-Hall International Editions, 2ª edición, 1992.

[OGATA-93] K. Ogata, "Modelado Matemático de Sistemas Dinámicos", en: *Ingeniería de Control Moderna*, Prentice-Hall Hispanoamericana, 2ª edición,1993.

[PAUL-81] R.P. Paul, "Dynamics", en: *Robot Manipulators, Mathematics, Programming and Control*, The MIT Press, Massachussets, 1982.

[SALTAREN-96] R. Saltaren, *Herramientas para el modelado y diseño paramétrio de robots basadas en la dinámica de los sistemas multicuerpo*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1996

[UICKER-65] J.J.Uicker, *On the Dynamic Analysis of Spacial Linkages Using 4x4 Matrices*, Ph.D. Dissertation, Northwestern University, Evaston, Ill, 1965.

[UICKER-64] J.J Uicker, J. Denavit y R.S. Hartenberg, "An Iterative Method for the Displacement Analysis of Spatial Mechanism", *Trans. ASME Journal of Applied Mech.*, Vol. 31, Serie B, 1964.

[WATTON-89], J. Watton, "Response, Stability and Control of Electrohydraulics Systems", en: *Fluid Power Systems*, Prentice-Hall, 1989.

Capítulo 6

CONTROL CINEMÁTICO

No hay atajo sin trabajo.
Refranero popular

En capítulos anteriores se ha visto cómo obtener el modelo cinemático y dinámico de un robot. El objetivo fundamental de ambos modelos es el de poder establecer las adecuadas estrategias de control del robot que redunden en una mayor calidad de sus movimientos.

El control cinemático establece cuáles son las trayectorias que debe seguir cada articulación del robot a lo largo del tiempo para lograr los objetivos fijados por el usuario (punto de destino, trayectoria cartesiana del efector final del robot, tiempo invertido por el usuario, etc.). Estas trayectorias se seleccionarán atendiendo a las restricciones físicas propias de los accionamientos y a ciertos criterios de calidad de trayectoria, como suavidad o precisión de la misma.

6.1. FUNCIONES DE CONTROL CINEMÁTICO

La Figura 6.1 muestra de manera esquemática el funcionamiento del control cinemático. Recibe como entradas los datos procedentes del programa del robot escrito por el usuario (punto de destino, precisión, tipo de trayectoria deseada, velocidad o tiempo invertido, etc.) y apoyándose en el modelo cinemático del robot establece las trayectorias para cada articulación como funciones de tiempo. Estas trayectorias deben ser muestreadas con un período T a decidir, generándose en cada instante kT un vector de referencias articulares para los algoritmos de control dinámico.

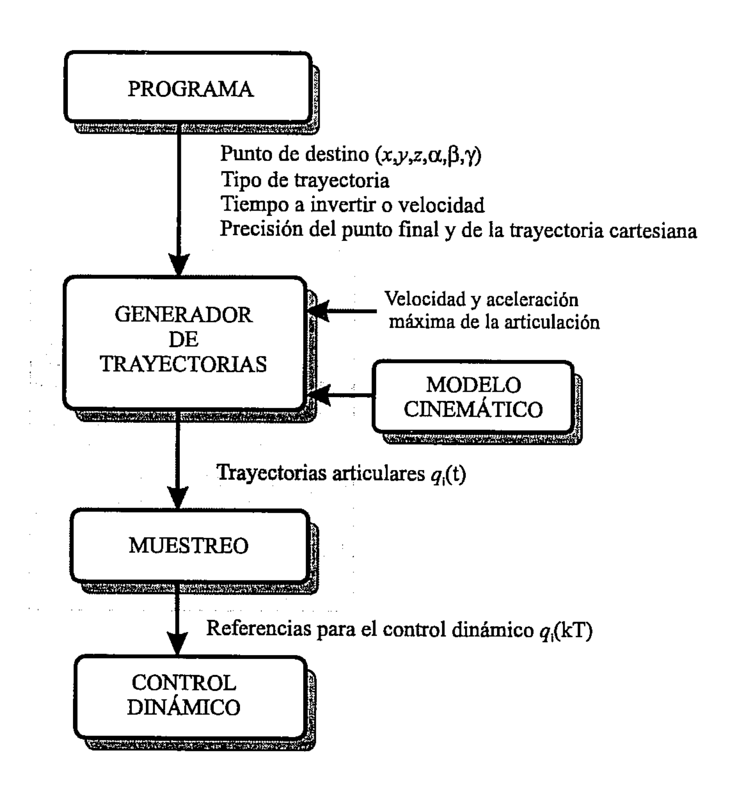


Figura 6.1. Funcionamiento del control cinemático (en sombreado).

De manera general, el control cinemático deberá realizar las siguientes funciones:

1. Convertir la especificación del movimiento dada en el programa en una trayectoria analítica en espacio cartesiano (evolución de cada coordenada cartesiana en función del tiempo).
2. Muestrear la trayectoria cartesiana obteniendo un número finito de puntos de dicha trayectoria. Cada uno de estos puntos vendrá dado por una 6-upla, típicamente $(x, y, z, \alpha, \beta, \gamma,)$.
3. Utilizando la transformación homogénea inversa, convertir cada uno de estos puntos en sus correspondientes coordenadas articulares $(q_1,q_2,q_3,q_4,q_5,q_6)$. Debe tenerse en cuenta aquí la posible solución múltiple de la transformación homogénea inversa, así como la posibilidad de ausencia de solución y puntos singulares, de modo que se asegure la continuidad de la trayectoria.
4. Interpolación de los puntos articulares obtenidos, generando para cada variable articular una expresión q_i(t) que pase o se aproxime a ellos de modo que, siendo una trayectoria realizable por los actuadores, se transforme en una trayectoria cartesiana lo más próxima a la especificada por el programa del usuario (en cuanto a precisión, velocidad, etc.).
5. Muestreo de la trayectoria articular para generar referencias al control dinámico.

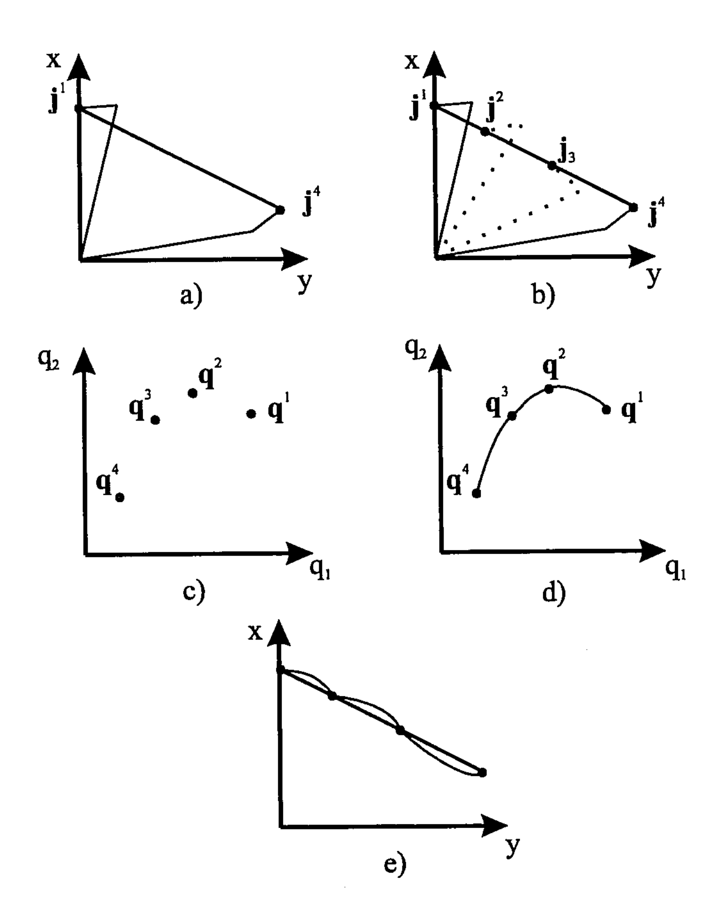


Figura 6.2. Ejemplo de control cinemático de un robot de 2 grados de libertad.

La Figura 6.2 muestra un posible caso en el que se pretende que un robot de 2 grados de libertad se mueva en línea recta desde el punto $\mathbf{j}^1=(x^1, y^1)$ hasta el punto $\mathbf{j}^4=(x^4, y^4)$ en un tiempo T (Figura 6.2-a). El control cinemático selecciona 4 puntos de esta trayectoria $\mathbf{j}^1$, $\mathbf{j}^2$, $\mathbf{j}^3$, $\mathbf{j}^4$ (Figura 6.2-b) y mediante la transformación homogénea inversa obtiene los correspondientes vectores articulares $\mathbf{q}^1$, $\mathbf{q}^2$, $\mathbf{q}^3$, $\mathbf{q}^4$ (Figura 6.2-c). A continuación trata de unir estos 4 puntos con algún tipo de función que pasando por todos ellos garantice la suavidad y no supere las velocidades y aceleraciones máximas permisibles para cada accionador (Figura 6.2-d). El resultado final del movimiento del extremo del robot es una trayectoria que se aproxima en mayor o menor medida a la línea recta deseada (Figura 6.2-e).

El principal inconveniente del procedimiento descrito para la generación de trayectorias radica en la necesidad de resolver repetidamente la transformación homogénea inversa, lo que conlleva un elevado coste computacional. Como alternativa, puede ser considerado un procedimiento basado en la utilización de la matriz Jacobiana.

Como se describió en el apartado 4.3, la matriz Jacobiana establece las relaciones diferenciales entre variables articulares y cartesianas en términos de la ecuación [4.60], que de manera compacta puede expresarse como:

$$\dot{\mathbf{j}}(t) = \mathbf{J}(q)\,\dot{\mathbf{q}}(t) \qquad [6.1]$$

Esta relación es lineal y para una localización **q** determinada permite conocer las velocidades cartesianas a partir de las correspondientes articulares. Además, suponiendo $\mathbf{J}(q)$ invertible, lo que ocurrirá siempre que las dimensiones del espacio articular y de la tarea sean iguales y **q** no sea un punto singular, se podrá expresar:

$$\dot{\mathbf{q}}(t) = \mathbf{J}^{-1}(q)\dot{\mathbf{j}}(t) \qquad [6.\ 2]$$

De este modo, a partir de la trayectoria cartesiana deseada para el extremo del robot **j**(t) podrá obtenerse su derivada y, a través de la relación anterior, la velocidad que debe seguir cada una de las articulaciones. Debe considerarse que esta relación es lineal y, por lo tanto más, fácil de obtener que la que utiliza la transformación homogénea. No hay que olvidar, sin embargo, que $\mathbf{J}(q)$ varía para cada localización **q** considerada, siendo por tanto necesario actualizar continuamente los valores de los elementos de la matriz **J**.

La implementación de este procedimiento en la unidad de control de un robot exigirá, una vez más, la discretización de la trayectoria **j**(t) y su derivada en un número finito de puntos que, posteriormente, y tras ser convertidos a velocidades articulares deberán ser interpolados.

Por lo tanto, para estudiar el control cinemático de un robot se deberá fundamentalmente conocer qué tipo de interpoladores pueden ser eficaces para unir los puntos articulares por los que se quiere pasar. Asimismo, será conveniente establecer criterios para seleccionar cuántos y qué puntos serán muestreados en la trayectoria del espacio cartesiano.

6.2. TIPOS DE TRAYECTORIAS

Para realizar una tarea determinada el robot debe moverse desde un punto inicial a un punto final. Este movimiento puede ser realizado según infinitas trayectorias espaciales. De todas ellas hay algunas que, bien por su sencillez de implementación por parte del control cinemático o bien por su utilidad y aplicación a diversas tareas, son las que en la práctica incorporan los robots comerciales. De este modo, puede encontrarse que los robots dispongan de trayectorias punto a punto, coordinadas y continuas.

6.2.1. Trayectorias punto a punto

En este tipo de trayectorias cada articulación evoluciona desde su posición inicial a la final sin realizar consideración alguna sobre el estado o evolución de las demás articulaciones. Normalmente, cada actuador trata de llevar a su articulación al punto de destino en el menor tiempo posible, pudiéndose distinguir dos casos: movimiento eje a eje y movimiento simultáneo de ejes.

Movimiento eje a eje

Sólo se mueve un eje cada vez. Comenzará a moverse la primera articulación, y una vez que ésta haya alcanzado su punto final lo hará la segunda, y así sucesivamente. Este tipo de movimiento da obviamente como resultado un mayor tiempo de ciclo, teniendo como única ventaja un menor consumo de potencia instantánea por parte de los actuadores.

Movimiento simultáneo de ejes

En este caso todos los actuadores comienzan simultáneamente a mover las articulaciones del robot a una velocidad específica para cada una de ellas. Dado que la distancia a recorrer y las velocidades serán en general diferentes, cada una acabará su movimiento en un instante diferente.

El movimiento del robot no acabará hasta que se alcance definitivamente el punto final, lo que se producirá cuando el eje que más tarde concluya su movimiento. De esta manera, el tiempo total invertido en el movimiento coincidirá con el del eje que más tiempo emplee en realizar su movimiento particular, pudiéndose dar la circunstancia de que el resto de los actuadores hayan forzado su movimiento a una velocidad y aceleración elevada, viéndose obligados finalmente a esperar a la articulación más lenta.

Por los motivos expuestos, las trayectorias punto a punto no están implementadas salvo en robots muy simples o con unidades de control muy limitadas.

6.2.2. Trayectorias coordinadas o isocronas

Para evitar que algunos actuadores trabajen forzando sus velocidades y aceleraciones, teniendo que esperar después la conclusión del movimiento de la articulación más lenta, puede hacerse un cálculo previo, averiguando cuál es esta articulación y qué tiempo invertirá. Se ralentizará entonces el movimiento del resto de los ejes para que inviertan el mismo tiempo en su movimiento, acabando todos ellos simultáneamente. Se tiene así que todas las articulaciones se coordinan comenzando y acabando su movimiento a la vez, adaptándose todas a la más lenta.

El tiempo total invertido en el movimiento es el menor posible y no se piden aceleraciones y velocidades elevadas a los actuadores de manera inútil. Desde el punto de vista del usuario la trayectoria que describe el extremo del robot no es significativa, siendo ésta impredecible, aunque como es obvio, un conocimiento del modelo y control cinemático del robot permitiría su cálculo.

6.2.3. Trayectorias continuas

Cuando se pretende que la trayectoria que sigue el extremo del robot sea conocida por el usuario (trayectoria en el espacio cartesiano o de la tarea), es preciso calcular de manera continua las trayectorias articulares.

Típicamente, las trayectorias que el usuario pretende que el robot describa son trayectorias en línea recta o en arco de círculo. Para conseguirlo habrá que seguir los pasos indicados en el epígrafe 6.1. El resultado será que cada articulación sigue un movimiento aparentemente caótico con posibles cambios

de dirección y velocidad y sin coordinación con el resto de las articulaciones. Sin embargo, el resultado conjunto será que el extremo del robot describirá la trayectoria deseada.

La Figura 6.3 representa las trayectorias articulares q_1(t) y q_2(t) y el resultado final en el espacio de la tarea (x, y) para un robot tipo SCARA correspondientes a los cuatro tipos de trayectorias indicadas.

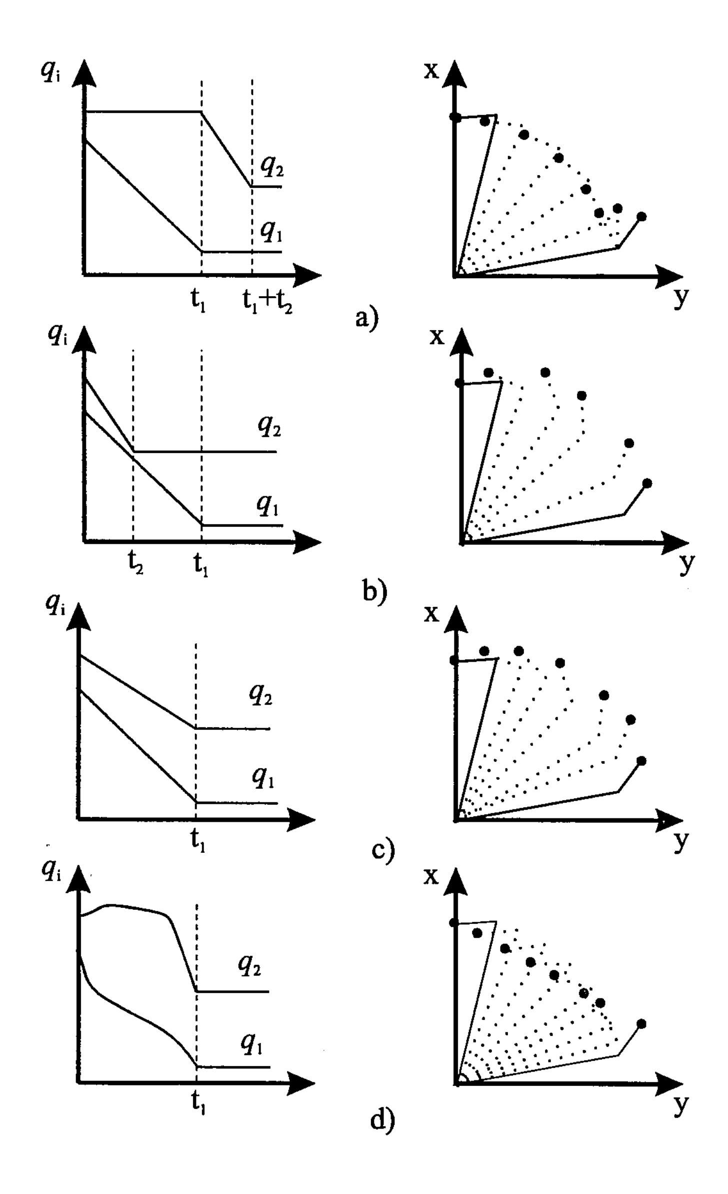


Figura 6.3. Diferentes trayectorias articulares posibles para un robot SCARA de 2 GDL. a) movimiento eje a eje, b) movimiento simultáneo de ejes, c) trayectoria coordinada, d) trayectoria continua rectilínea.

6.3. GENERACIÓN DE TRAYECTORIAS CARTESIANAS

Normalmente el usuario del robot indica el movimiento que éste debe realizar especificando las localizaciones espaciales por las que debe pasar el extremo, junto con otros datos, como instantes de paso, velocidades o tipos de trayectorias. Así, por ejemplo, es frecuente especificar que el robot debe ir de un punto inicial hasta otro final, siguiendo en cartesianas una línea recta a velocidad constante.

Puesto que estos puntos están excesivamente separados (salvo tal vez la excepción de algunos robots de pintura programados por guiado, tal y como se verá en el Capítulo 8) es preciso seleccionar puntos intermedios suficientemente cercanos como para que el control del robot consiga ajustar no sólo el punto final al especificado, sino también la trayectoria seguida a la indicada en el programa.

Para ello es preciso establecer un interpolador entre las localizaciones expresadas en el espacio de la tarea que dará como resultado una expresión analítica de la evolución de cada coordenada. La interpolación más frecuente es la lineal, en la que cada coordenada evoluciona a velocidad constante desde su valor inicial j^i hasta el final j^f:

$$j(t) = (j^f - j^i)\frac{t - t_i}{t_f - t_i} + j^i \qquad [6.3]$$

donde t_i y t_f son los instantes de tiempo en los que se pretende alcanzar la localización inicial y final, respectivamente.

Para evitar las discontinuidades de velocidad en el caso de paso por varios puntos, pueden utilizarse las técnicas de interpoladores a tramos o interpoladores cúbicos que se describen en el epígrafe 6.4 para el caso de variables articulares.

6.3.1. Evolución de la orientación

Como es conocido, la especificación de la posición por parte del usuario se realiza habitualmente, y salvo escasas excepciones, en coordenadas cartesianas. Sin embargo, tal y como se vio en el Capítulo 3, la especificación de la orientación puede realizarse mediante diferentes herramientas, como son: matrices de rotación, ángulos de Euler (en sus diferentes versiones), par de rotación o cuaternios.

Esta única alternativa de representación de la posición no deja dudas en cuanto a que el procedimiento más adecuado para unir dos posiciones en el espacio de la tarea es la evolución lineal, a velocidad constante, de cada coordenada cartesiana desde su valor incial hasta el final, resultando un movimiento sencillo, fácilmente interpretable por el usuario y de calidad.

Sin embargo, para el caso de la orientación, esta evolución lineal desde el valor inicial hasta el final puede ser planteado en términos de matrices de rotación, ángulos de Euler, par de rotación o cuaternios, resultando en cada caso trayectorias diferentes.

La utilización de las matrices de rotación lleva a resultados inconsistentes. Como es sabido, las matrices de rotación deben ser necesariamente matrices ortonormales. La interpolación lineal entre

una matriz de rotación inicial y otra final lleva a matrices intermedias no ortonormales y, por lo tanto, que no corresponden a matrices de rotación.

Ejemplo 6.1.

Supóngase que se quiere evolucionar desde una orientación inicial dada por:

$$\mathbf{R}_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

hasta otra final dada por:

$$\mathbf{R}_f = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$$

La orientación $\mathbf{R}_f$ corresponde a una rotación desde la orientación definida por $\mathbf{R}_i$ de 90° respecto del eje **z** seguido de una rotación de 90° respecto del eje **x**.

Como se puede comprobar, las columnas o filas de ambas matrices son ortonormales. La evolución lineal entre estas 2 matrices daría lugar a una a una matriz intermedia dada por:

$$\mathbf{R}_m = \begin{bmatrix} 1/2 & -1/2 & 0 \\ 0 & 1/2 & -1/2 \\ 1/2 & 0 & 1/2 \end{bmatrix}$$

Cuyas filas y columnas no son ortonormales y, por lo tanto, $\mathbf{R}_m$ no puede corresponder a una orientación válida del extremo del robot.

La utilización de ángulos de Euler, en cualquiera de sus formas, además de ser la representación más compacta, no presenta este inconveniente. Así, para pasar de una orientación inicial (α_i, β_i, γ_i) a una final (α_f, β_f, γ_f) se podrían utilizar las funciones lineales:

$$\alpha(t) = (\alpha_f - \alpha_i)\frac{t - t_i}{t_f - t_i} + \alpha_i$$

$$\beta(t) = (\beta_f - \beta_i)\frac{t - t_i}{t_f - t_i} + \beta_i \qquad [6.4]$$

$$\gamma(t) = (\gamma_f - \gamma_i)\frac{t - t_i}{t_f - t_i} + \gamma_i$$

donde t_i y t_f son los instantes de tiempo en los que se pretende estar en la orientación inicial y final, respectivamente. El inconveniente de esta trayectoria es que desde el punto de vista del usuario, es una trayectoria no intuitiva, con extrañas evoluciones de la orientación.

La evolución más natural desde una orientación inicial hasta otra final, sería aquella que hace girar de manera progresiva al efector final (u objeto manipulado por el robot) desde su orientación inicial hasta la final en torno a un eje de giro fijo. Por este motivo, la utilización del par de rotación, o su equivalente, los cuaternios, es la manera más adecuada para generar la trayectoria cartesiana de orientación.

Como se estableció en el epígrafe 3.2.3, dado un sistema ortonormal inicial y otro final rotado respecto del primero, existe un único eje **k** que permite pasar del sistema inicial al final girando un ángulo θ respecto a él. Por lo tanto, para que el extremo del robot evolucione desde la orientación inicial hasta la final, se podrá buscar cuál es el par (**k**,θ) que relaciona los sistemas de coordenadas ortonormales asociados a ambas orientaciones, y realizar la evolución temporal mediante un giro en torno al eje **k** de valor:

$$\theta(t) = \theta\frac{t - t_i}{t_f - t_i} \qquad [6.5]$$

A partir del valor de θ(t) para instantes concretos de tiempo será inmediato conocer el cuaternio correspondiente.

6.4. INTERPOLACIÓN DE TRAYECTORIAS

Como se ha indicado, una de las funciones del control cinemático es la de unir una sucesión de puntos en el espacio articular por los que se quiere que pasen las articulaciones del robot en un instante determinado. Además, junto con las condiciones de posición-tiempo, es conveniente añadir restricciones en la velocidad y aceleración de paso por los puntos, de manera que se asegure la suavidad de la trayectoria y se limiten las velocidades y aceleraciones máximas. Estas restricciones garantizarán que los actuadores están capacitados para implementar la trayectoria final.

Para ello deberá seleccionarse algún tipo de función (frecuentemente polinómica) cuyos parámetros o coeficientes se ajustarán al imponer las condiciones de contorno: posiciones, velocidades y aceleraciones. En la selección de esta función debe considerarse que tanto el cálculo de sus parámetros, como su posterior utilización para generar puntos de consigna al control dinámico, debe hacerse en tiempo real, por lo que la simplicidad de la función será un factor a valorar.

Se van a presentar a continuación las funciones interpoladoras utilizadas con mayor frecuencia. Cada una de ellas ha sido desarrollada para un solo grado de libertad, debiendo quedar claro que el mismo cálculo deberá repetirse para cada uno de los grados de libertad del robot. Cabe indicar que si bien las técnicas de interpolación que a continuación se describen están planteadas para el espacio articular son igualmente aplicables para el espacio de la tarea.

6.4.1. Interpoladores lineales

Supóngase que se pretende que una de las articulaciones q del robot, pase sucesivamente por los valores q^i en los instantes t^i. Una primera solución a este problema consistiría en mantener constante la velocidad de movimiento entre cada 2 valores sucesivos (q^{i-1}, q^i) de la articulación.

La trayectoria entre dos puntos q^{i-1}, q^i sería entonces:

$$q(t) = \left(q^i - q^{i-1}\right)\frac{t - t^{i-1}}{T} + q^{i-1} \qquad t^{i-1} < t < t^i$$
$$T = t^i - t^{i-1} \qquad [6.6]$$

Como es obvio, esta trayectoria asegura la continuidad de la posición, pero origina saltos bruscos en la velocidad $\dot{q}$ de la articulación, y consecuentemente precisa de aceleraciones $\ddot{q}$ de valor infinito (ver Figura 6.4), lo que en la práctica no es posible.

La selección de los instantes de paso t^i por los puntos q^i podrá haberse hecho según los diferentes criterios expuestos en el epígrafe 6.2. Esto es:

1. Intentando que cada articulación q alcance el punto de destino en el menor tiempo posible sin considerar las demás articulaciones, lo que resultará en velocidades $\dot{q}$ constantes e iguales a la máxima.
2. Ajustando los instantes de paso a los de la articulación que más tiempo precise, resultando movimientos coordinados.
3. Seleccionando los tiempos a partir de las especificaciones dadas en el espacio de la tarea de modo que el extremo del robot describa una trayectoria predeterminada.

6.4.2. Interpoladores cúbicos

Para asegurar que la trayectoria que une los puntos por los que tiene que pasar la articulación considerada presente continuidad en velocidad, puede recurrirse a utilizar un polinomio de grado 3 que una cada pareja de puntos adyacentes. De este modo, al tener cuatro parámetros disponibles se podrán imponer cuatro condiciones de contorno, dos de posición y dos de velocidad. Los valores de las

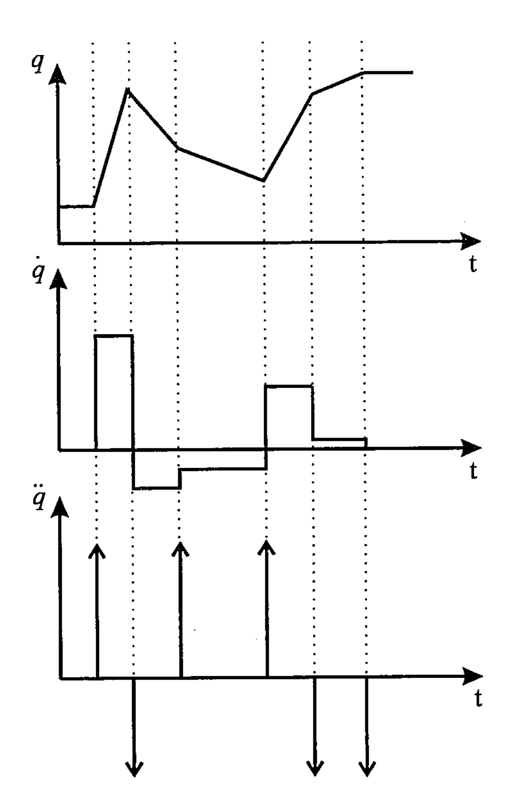


Figura 6.4. Posición, velocidad y aceleración para un interpolador lineal.

velocidades de paso por cada punto deben por tanto ser conocidas *a priori*. Esta selección no es obvia y se tratará posteriormente.

Se consigue así una trayectoria compuesta por una serie de polinomios cúbicos, cada uno válido entre dos puntos consecutivos. Este conjunto de polinomios concatenados, escogidos de modo que exista continuidad en la posición y velocidad, se denominan **splines** (cúbicos, por ser de tercer grado). Una exposición detallada de los mismos puede encontrarse en [ANAND-93].

La expresión de la trayectoria que une dos puntos adyacentes (q^{i-1}, q^{i}) será:

$$
\begin{aligned}
& q(\mathrm{t}) = \mathrm{a} + \mathrm{b}\left(\mathrm{t}-\mathrm{t}^{i-1}\right) + \mathrm{c}\left(\mathrm{t}-\mathrm{t}^{i-1}\right)^2 + \mathrm{d}\left(\mathrm{t}-\mathrm{t}^{i-1}\right)^3 \qquad \mathrm{t}^{i-1} < \mathrm{t} < \mathrm{t}^{i} \\
& \mathrm{a} = q^{i-1} \\
& \mathrm{b} = \dot{q}^{i-1} \\
& \mathrm{c} = \frac{3}{\mathrm{T}^2}\left(q^{i} - q^{i-1}\right) - \frac{2}{\mathrm{T}^2}\dot{q}^{i-1} - \frac{1}{\mathrm{T}^2}\dot{q}^{i} \\
& \mathrm{d} = -\frac{2}{\mathrm{T}^3}\left(q^{i} - q^{i-1}\right) + \frac{1}{\mathrm{T}^2}\left(\dot{q}^{i-1} + \dot{q}^{i}\right) \\
& \mathrm{T} = \mathrm{t}^{i} - \mathrm{t}^{i-1}
\end{aligned}
\qquad [6.\ 7]
$$

Para poder calcular los valores de los coeficientes del polinomio cúbico de la expresión [6.7] es preciso conocer los valores de las velocidades de paso $\dot{q}^i$. Para ello existirán diferentes alternativas.

Un primer criterio para seleccionar las velocidades de paso [CRAIG-89] podría ser:

$$\dot{q}^{i} = \begin{cases} 0 & \text{si signo}\left(q^{i}-q^{i-1}\right) \neq \text{signo}\left(q^{i+1}-q^{i}\right) \\ \dfrac{1}{2}\left[\dfrac{q^{i+1}-q^{i}}{t^{i+1}-t^{i}}+\dfrac{q^{i}-q^{i-1}}{t^{i}-t^{i-1}}\right] & \text{si signo}\left(q^{i}-q^{i-1}\right) = \text{signo}\left(q^{i+1}-q^{i}\right) \end{cases} \qquad [6.\ 8]$$

Esta selección es de sencilla realización y da como resultado una continuidad razonable en la velocidad. Sin embargo, no establece ninguna condición sobre la continuidad de la aceleración.

Como alternativa pueden escogerse las velocidades de paso de modo que cada spline cúbico sea continuo en posición, velocidad y aceleración con los dos polinomios adyacentes. De esta forma, los coeficientes de los k-1 polinomios de tipo spline cúbico que pasan por los puntos q^i ($i \in [1,k]$), asegurando la continuidad en posición, velocidad y aceleración de la trayectoria global, serán los dados por la expresión [6.7], donde las velocidades de paso por los puntos se obtienen de resolver el siguiente sistema de ecuaciones lineales de diagonal dominante [ROGERS-76]:

$$\begin{bmatrix} t^3 & 2(t^2+t^3) & t^2 & 0 & 0 & \dots \\ 0 & t^4 & 2(t^3+t^4) & t^3 & 0 & \dots \\ 0 & 0 & t^5 & 2(t^4+t^5) & t^4 & \dots \\ \dots & \dots & 0 & t^6 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \cdot \begin{bmatrix} \dot{q}^1 \\ \dot{q}^2 \\ \dot{q}^3 \\ \vdots \\ \dot{q}^k \end{bmatrix} = \begin{bmatrix} \dfrac{3}{t^2 t^3}\left[(t^2)^2(q^3-q^2)+(t^3)^2(q^2-q^1)\right] \\ \dfrac{3}{t^3 t^4}\left[(t^3)^2(q^4-q^3)+(t^4)^2(q^3-q^2)\right] \\ \vdots \\ \dfrac{3}{t^{k-1} t^k}\left[(t^{k-1})^2(q^k-q^{k-1})+(t^k)^2(q^{k-1}-q^{k-2})\right] \end{bmatrix} \qquad [6.\ 9]$$

Como se observa, el sistema tiene k-2 ecuaciones y k incógnitas, que son las distintas velocidades de paso por los k puntos. Para completar el número de ecuaciones de modo que el sistema esté definido, se pueden añadir las siguientes condiciones:

$$\dot{q}^1 = \dot{q}^k = 0 \qquad [6.\ 10]$$

Es decir, que la articulación parte de y llega a una situación de reposo. Por lo tanto, las k ecuaciones lineales definidas en [6.9] y [6.10] permiten obtener las k velocidades de paso necesarias para poder aplicar la expresión [6.7] de los splines cúbicos, asegurando la continuidad hasta la segunda derivada de la trayectoria global.

Una última alternativa para obtener las velocidades de paso $\dot{q}^i$ sería partir de las velocidades de paso deseadas en el espacio de la tarea. De este modo, lo mismo que el modelo geométrico del robot permite obtener las coordenadas articulares q a partir de las cartesianas j, el modelo cinemático (mediante el uso de la Jacobiana) permitiría obtener las velocidades articulares $\dot{q}$ a partir de las carte-

sianas $\dot{j}$. Sin embargo, es preciso en este caso considerar los problemas derivados de la posible existencia de puntos singulares en la trayectoria cartesiana escogida, que darían lugar a velocidades articulares infinitas no realizables por los actuadores.

6.4.3. Interpoladores a tramos

En los interpoladores vistos hasta el momento, se utiliza un polinomio de un grado determinado (1 o 3) para unir dos puntos consecutivos de la trayectoria. El uso de polinomios de tercer grado permite asegurar que el polinomio pasa por los dos puntos y al mismo tiempo permite imponer los valores de velocidad de paso por los mismo. Sin embargo, al contrario de lo que ocurre con el interpolador de primer grado (lineal), la velocidad de la articulación durante el recorrido está variando continuamente, lo que exige un control continuo de la misma.

Una alternativa que proporciona una solución intermedia consiste en descomponer en tres tramos consecutivos la trayectoria que une dos puntos q^0, q^1. En el tramo central se utiliza un interpolador lineal, y por lo tanto la velocidad se mantiene constante, no siendo preciso imprimir aceleración alguna al actuador. En los tramos inicial y final se utiliza un polinomio de segundo grado, de modo que en el tramo 1 la velocidad varíe linealmente desde la velocidad de la trayectoria anterior a la de la pre-

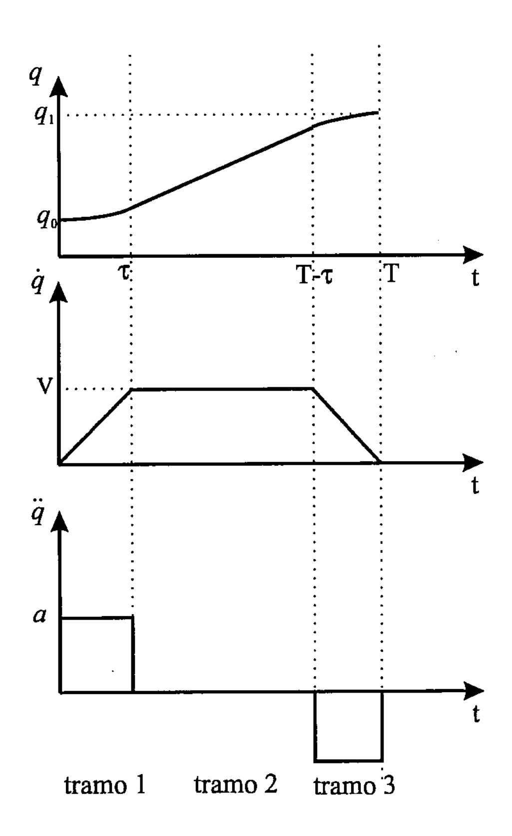


Figura 6.5. Interpolador con tres tramos.

sente, y en el tramo 3 varíe desde la velocidad de la trayectoria presente hasta la de la siguiente. Se tiene entonces que en los tramos inicial y final la aceleración toma valores constantes distintos de cero, mientras que en el tramo intermedio la aceleración es nula.

En el caso simple de una trayectoria con dos únicos puntos de velocidad inicial y final nula, se tiene (ver Figura 6.5) que las ecuaciones de los tres tramos serían:

$$q(t) = \begin{cases} q^0 + s\frac{a}{2}t^2 & t \leq \tau \\ q^0 - s\frac{V^2}{2a} + sVt & \tau < t \leq T - \tau \\ q^1 + s\left(-\frac{aT^2}{2} + aTt - \frac{a}{2}t^2\right) & T - \tau < t < T \end{cases} \qquad [6.\ 11]$$

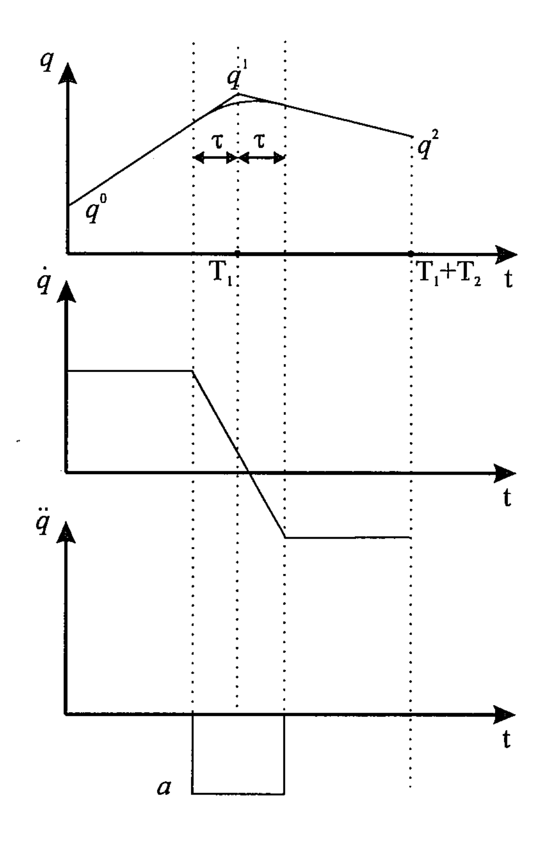


Figura 6.6. Interpolador con ajuste parabólico.

siendo :

$$\tau = \frac{V}{a}$$

$$T = s\frac{q^1 - q^0}{V} + \frac{V}{a}$$

V: velocidad máxima permitida

a: aceleración a utilizar

s: signo $\left(q^1 - q^0\right)$

En el caso de tener una trayectoria formada por varios puntos, la velocidad de paso por los puntos intermedios no debería ser nula, puesto que esto daría lugar a un movimiento discontinuo del robot. Esta situación puede ser evitada si se permite que la trayectoria no pase exactamente por los puntos. La trayectoria final coincidiría con las trayectorias rectilíneas que unen los puntos dos a dos salvo en las cercanías de los mismos, donde un polinomio de 2° grado permitiría variar progresivamente la velocidad, evitando los valores infinitos de la aceleración que han sido descritos en el apartado 6.4.1. Ver la Figura 6.6.

Esta técnica, denominada ajuste parabólico, se aproxima al interpolador lineal tanto más cuanto mayor sea la aceleración permitida, coincidiendo con aquél en el caso de aceleración infinita. Así, si se quiere pasar por los puntos q^0, q^1 y q^2 en los instantes $t = 0$, $t = T_1$ y $t = T_1 + T_2$, respectivamente, se tendría que la ecuación de los tres tramos que componen la trayectoria que une dos puntos consecutivos sería:

$$q(t) = \begin{cases} q^0 + \dfrac{q^1 - q^0}{T_1} t & 0 \le t \le T_1 - \tau \\ q^1 + \dfrac{\left(q^1 - q^0\right)}{T_1}\left(t - T_1\right) + \dfrac{a}{2}\left(t - T_1 + \tau\right)^2 & T_1 - \tau < t < T_1 + \tau \\ q^1 + \dfrac{q^2 - q^1}{T_2}\left(t - T_1\right) & T_1 + \tau < t < T_1 + T_2 \end{cases} \qquad [6.\ 12]$$

Donde a es la aceleración constante que permite cambiar la velocidad de un tramo al siguiente, siendo su valor:

$$a = \frac{T_1\left(q^2 - q^1\right) - T_2\left(q^1 - q^0\right)}{2T_1T_2\tau} \qquad [6.\ 13]$$

y 2τ es el tiempo utilizado en variar la velocidad del movimiento, estando repartido simétricamente respecto al instante T_1. Lógicamente, cuanto mayor sea la aceleración *a* permitida menor será el tiempo de transición τ.

Asimismo, es fácil obtener el máximo error cometido entre la trayectoria ideal (pasando por q^1) y la real con el segmento parabólico. Este se produce en $t = T_1$ y toma el valor:

$$e = \frac{a}{2}\tau^2 = \frac{T_1(q^2 - q^1) - T_2(q^1 - q^0)}{4T_1T_2}\tau \qquad [6.\ 14]$$

Este error disminuye al permitir tiempos de cambio de trayectoria τ más pequeños, siendo nulo si no se utilizase tramo parabólico ($\tau = 0$) y la aceleración *a* pudiese tomar el valor infinito.

6.4.4. Otros interpoladores

Diferentes tipos de interpoladores han sido propuestos y utilizados para conseguir unir los puntos de la trayectoria con funciones continuas, derivables dos veces y computacionalmente manejables. Aparte de las ya citadas, puede destacarse la utilización de funciones senoidales , que aseguran continuidad en todas las derivadas. De este modo, para unir dos puntos consecutivos podrían utilizarse funciones de la forma:

$$q(t) = a + bt + c\,\text{sen}(\omega t) \qquad [6.\ 15]$$

Imponiendo las condiciones de contorno en posición, velocidad y aceleración, se obtendrían los correspondientes coeficientes. En [MUJTABA-77] puede verse un estudio comparativo de diversos tipos de interpoladores.

6.5. MUESTREO DE TRAYECTORIAS CARTESIANAS

Una vez analizadas algunas de las diferentes alternativas para interpolar los puntos seleccionados de la trayectoria que garanticen la obtención de curvas realizables por los actuadores, queda por establecer criterios adecuados para la selección óptima de los puntos de la trayectoria cartesiana.

La trayectoria cartesiana más habitual es la línea recta, utilizándose también en ciertas ocasiones trayectorias circulares. Ambas pueden ser descritas con facilidad de manera analítica en función del tiempo, de modo que es fácil conocer las coordenadas cartesianas por las que se desea que pase el extremo del robot en un instante concreto.

Pudiera parecer en principio que cuanto mayor sea el número de instantes en los que se muestrea la trayectoria cartesiana, se obtendrán mejores resultados, por cuanto que la trayectoria final pasaría por estos puntos ajustándose a lo especificado con mayor precisión. Sin embargo, debe considerarse que

cada uno de estos puntos debe ser transformado a sus correspondientes coordenadas articulares y ser después utilizado en alguno de los interpoladores presentados en el epígrafe 6.3 para generar la trayectoria articular. Por lo tanto es claro que el costo computacional, limitado por la necesaria respuesta en tiempo real del sistema de control del robot, desaconseja que el número de puntos cartesianos a tomar crezca de manera indiscriminada.

Por otra parte, la no linealidad existente en el modelo geométrico del robot indica que, dependiendo de su cinemática y de la posición instantánea en cuestión, el error cometido entre la trayectoria resultante del extremo del robot y la trayectoria cartesiana deseada (que coincidirán en los puntos seleccionados) puede variar enormemente. Por lo tanto, es conveniente decidir no sólo cuántos puntos se seleccionan de la trayectoria cartesiana, sino también cuáles son estos puntos, teniendo siempre como objetivo llegar a un compromiso entre el número de puntos seleccionados y el error entre la trayectoria resultante y la deseada.

Con estos objetivos en [TAYLOR-79] se propone un algoritmo recursivo que permite seleccionar sólo aquellos puntos de la trayectoria cartesiana estrictamente necesarios, quedando éstos más próximos entre sí allí donde se produciría mayor error de la trayectoria, y más espaciados en los tramos en los que las trayectorias deseada y resultante son más cercanas.

En la práctica suele ser suficiente con seleccionar puntos equidistantes tan cercanos como sea posible, asegurando que las trayectorias articulares puedan ser generadas antes del tiempo en que sean requeridas por el control dinámico del robot.

6.6. BIBLIOGRAFÍA

[ANAND-93] V.B. Anand, "Curves", en : *Computer Graphics and Geometric Modeling for Engineers*, John Wiley & Sons, 1993.

[CRAIG-89] J.J. Craig, "Trayectory Generation", en: *Introduction to Robotics Mechanics and Control*, Addison-Wesley, 1989.

[FU-88] K.S. Fu, R.C. Gonzalez y C.S.G. Lee, "Planificación de Trayectorias de un Manipulador", en: *Robótica: Control, Detección, Visión e Inteligencia*, McGraw-Hill/Interamericana, Madrid, 1988.

[MUJTABA-77] M.S. Mujtaba, "Discussion of trayectory calculation methods", *Exploratory Study of Computer Integrated Assembly Systems*, Internal Report # STAN-CS-t6-568, Stanford Artificial Intelligent Laboratory, Stanford University, California.

[ROGERS-76] D.F.Rogers y J.A. Adams, "Space Curves", en: *Mathematical elements for computer graphics*, McGraw-Hill, 1976.

[SCHILLING-90] R.J. Schilling, *Fundamentals of Robotics Analisys and Control*, Prentice-Hall, 1990.

[TAYLOR-79] R.H. Taylor, "Planning and execution of straight line manipulator trajectories", *IBM Journal of Research and Development*, 23, pp. 424-426.

Capítulo 7

CONTROL DINÁMICO

Lo mejor es enemigo de lo bueno.
Refranero popular

A un robot se le demandan por lo general elevadas prestaciones en velocidad y precisión de movimiento. Este objetivo ha marcado en gran medida las investigaciones y avances tecnológicos realizados en el campo de la robótica. La utilización de materiales más ligeros, de sistemas de transmisión sin holgura y con baja inercia, de actuadores rápidos y precisos o de sensores de elevada resolución, han permitido mejorar de manera notable la calidad del movimiento del robot.

Del mismo modo, los algoritmos de control del robot tienen por objetivo mejorar al máximo las características de velocidad y precisión. El control cinemático selecciona trayectorias que idealmente deberá seguir el robot para, teniendo en cuenta sus limitaciones, ajustarse lo mejor posible a las especificaciones del movimiento dadas por el usuario. En la práctica, este ajuste del movimiento del robot a las especificaciones del usuario no será del todo posible, pues las características dinámicas del robot (inercias, rozamiento, holguras, etc.) muchas veces desconocidas, impiden en general una coincidencia entre la trayectoria deseada $\mathbf{q}_d(t)$ y la real $\mathbf{q}(t)$.

El **control dinámico** tiene por misión procurar que las trayectorias realmente seguidas por el robot $\mathbf{q}(t)$ sean lo más parecidas posibles a las propuestas por el control cinemático $\mathbf{q}_d(t)$. Para ello hace uso del conocimiento del modelo dinámico del robot (obtenido en el Capítulo 5) y de las herramientas de análisis y diseño aportadas por la teoría del servocontrol (representación interna, estado, estabilidad de Lyapunov, control PID, control adaptativo, etc.).

Como se vio al obtener el modelo dinámico de un robot, éste es fuertemente no lineal, multivariable, acoplado y de parámetros variantes, por lo que en general su control es extremadamente complejo. En la práctica ciertas simplificaciones, válidas para un gran número de los robots comerciales

existentes, facilitan el diseño del sistema de control, dando unos resultados razonablemente aceptables, aunque limitando en ciertas situaciones la calidad de sus prestaciones.

La utilización de técnicas de control más potentes, que podrían dar mejores resultados, pueden verse limitadas en la práctica por la elevada complejidad y coste computacional, que si bien es abordable con la tecnología actual, supone un mayor coste económico.

Normalmente el control dinámico se realiza en el espacio articular, esto es, controlando las trayectorias articulares $\mathbf{q}(t)$ del robot. Sin embargo, en ciertas ocasiones, como por ejemplo cuando el robot entra en contacto con el entorno desarrollando fuerzas de reacción, puede optarse por realizar el control en el espacio de la tarea o cartesiano, controlando la trayectoria del extremo $\mathbf{j}(t)$.

En este capítulo se van a examinar las técnicas de control dinámico más frecuentes, centrándose principalmente en el control en el espacio articular. Dentro de éste se considerará tanto el control monoarticular, en el que se desprecia la interacción entre los grados de libertad del robot, como el control multiarticular, en el que se considera al robot como el sistema multivariable que realmente es. Las técnicas de control utilizadas serán las basadas en control PID y control por prealimentación, ampliamente extendidas, utilizándose también en ocasiones la linealización por inversión del modelo. Se realizará asimismo alguna referencia a técnicas de control más potentes, como puede ser el control adaptativo.

7.1. CONTROL MONOARTICULAR

Las características mecánicas de un robot (dimensiones, peso, tipo de actuadores, etc.) influyen notablemente en el tipo de control a utilizar. Recordando una vez más la expresión del modelo dinámico de un robot (ecuación [5.16]):

$$\tau = \mathbf{D}(q)\ddot{\mathbf{q}} + \mathbf{H}(q,\dot{q}) + \mathbf{C}(q) + \mathbf{F}_v\dot{\mathbf{q}} \qquad [7.\ 1]$$

en la que se ha incluido la existencia del par debido al rozamiento viscoso mediante la matriz diagonal constante $\mathbf{F}_v$, puede intuirse cómo la estructura cinemática del robot va a modificar la forma de las matrices **D**, **H** y **C** y, por lo tanto, la complejidad del modelo y su correspondiente control.

Asimismo, factores como la presencia de reductores o la existencia de elevados rozamientos viscosos (notables, por ejemplo, al utilizar ciertos reductores o en los actuadores hidraúlicos) pueden hacer despreciables alguno de los términos de la ecuación [7.1].

7.1.1. Influencia del factor de reducción

Considérese por ejemplo que cada actuador incorpora un reductor, de modo que la posición de la articulación $\mathbf{q}$ está relacionada con la posición del eje del correspondiente actuador $\mathbf{q}_a$ mediante:

$$\mathbf{q}_a = \mathbf{K}\,\mathbf{q} \qquad [7.2]$$

donde **K** es la matriz diagonal de los factores de reducción, cuyos coeficientes son $k_{ii}>1$. Consecuentemente se verificarán las relaciones:

$$\dot{\mathbf{q}}_a = \mathbf{K}\,\dot{\mathbf{q}} \qquad \ddot{\mathbf{q}}_a = \mathbf{K}\,\ddot{\mathbf{q}} \qquad \tau_a = \mathbf{K}^{-1}\tau \qquad [7.3]$$

donde τ_a expresa el par resultante en el eje del actuador. Por otra parte, la matriz de inercia $\mathbf{D}(q)$ puede expresarse siempre como:

$$\mathbf{D}(q) = \mathbf{D}_1 + \mathbf{D}_2(q) \qquad [7.4]$$

donde $\mathbf{D}_1$ es una matriz diagonal formada por los elementos constantes de la diagonal principal de $\mathbf{D}(q)$ y representa la contribución de cada eslabón en la inercia que soporta su correspondiente actuador.

Sustituyendo las expresiones [7.3] y [7.4] en el modelo dinámico del robot [7.1], se obtiene:

$$\tau_a = \mathbf{K}^{-1}(\mathbf{D}_1 + \mathbf{D}_2)\mathbf{K}^{-1}\ddot{\mathbf{q}}_a + \mathbf{K}^{-1}\mathbf{H} + \mathbf{K}^{-1}\mathbf{C} + \mathbf{K}^{-1}\mathbf{F}_v\mathbf{K}^{-1}\dot{\mathbf{q}}_a \qquad [7.5]$$

que se puede expresar de la siguiente manera:

$$\tau_a = \mathbf{K}^{-1}\mathbf{D}_1\mathbf{K}^{-1}\ddot{\mathbf{q}}_a + \mathbf{F}_{va}\dot{\mathbf{q}}_a + \tau_p$$

donde:

$$\begin{aligned} \mathbf{F}_{va} &= \mathbf{K}^{-1}\mathbf{F}_v\mathbf{K}^{-1} \\ \tau_p &= \mathbf{K}^{-1}\mathbf{D}_2\mathbf{K}^{-1}\ddot{\mathbf{q}}_a + \mathbf{K}^{-1}\mathbf{H} + \mathbf{K}^{-1}\mathbf{C} \end{aligned} \qquad [7.6]$$

$\mathbf{F}_{va}$: es la matriz diagonal de constantes de rozamiento viscoso visto por los actuadores (a la entrada del reductor).

τ_p: puede ser considerado como un par perturbador.

La Figura 7.1 representa mediante diagrama de bloques las ecuaciones [7.6], donde los valores de los bloques representados por cajas ovaladas dependen de las variables que llegan a ellos a través de las flechas a tramos. Los bloques encerrados en el recuadro sombreado están formados por matrices diagonales constantes, lo que indica que representan el efecto que sobre el movimiento de la articulación i tiene el par proporcionado por el propio actuador. Por su parte, los bloques fuera del recuadro no son constantes sino que dependen de las posiciones y velocidades de los ejes y representan el efecto τ_p que sobre la articulación i tiene el movimiento de los demás ejes.

Este esquema permite evaluar en que condiciones puede considerarse al robot como a una serie de eslabones dinámicamente desacoplado (τ_p despreciable), en las que el movimiento de uno de ellos no afecta a los demás.

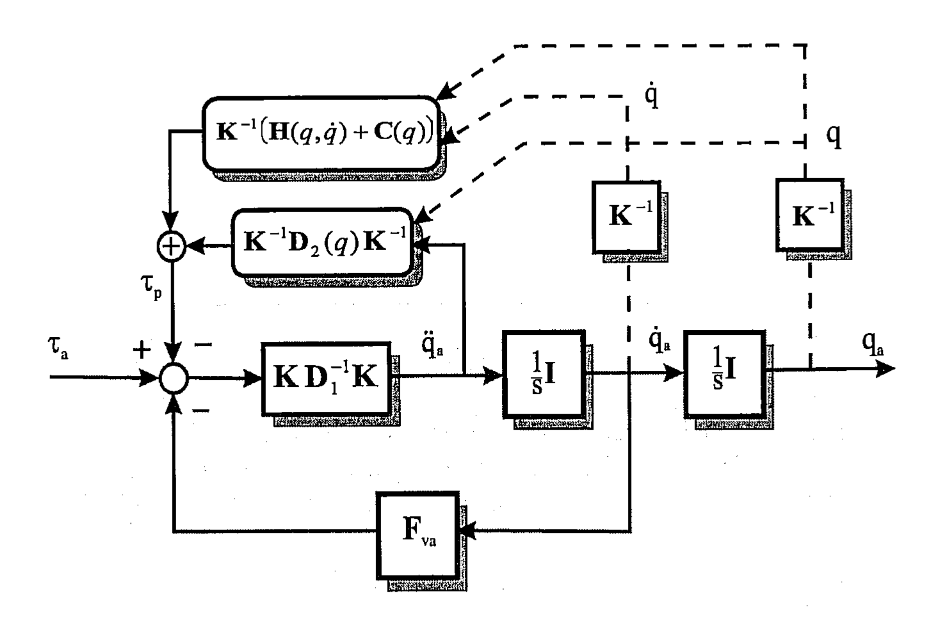


Figura 7.1. Diagrama de bloques que representa el comportamiento dinámico de una articulación.

Obsérvese que si los factores de reducción son elevados, esto es, si las velocidades del movimiento de los actuadores $\dot{q}_a$ son notablemente mayores que las de las articulaciones $\dot{q}$, se tendrá que los elementos de la matriz diagonal **K** serán muy superiores a la unidad, y consecuentemente K^{-1} será una matriz diagonal de elementos de reducido valor. En este caso, el término τ_p será despreciable y podrá considerarse que las articulaciones del robot están desacopladas, de modo que un par motor en el actuador i tendrá efecto únicamente sobre el movimiento de la articulación i, siendo esta relación la correspondiente a una ecuación diferencial lineal de 2º orden (recuérdese que **K**, D_l y F_{va} son matrices diagonales constantes). Por lo tanto, el diseño del regulador más adecuado para cada articulación i puede hacerse utilizando las técnicas más frecuentes de diseño (PID, prealimentación), cuyo objetivo es garantizar las especificaciones de funcionamiento aún ante perturbaciones o imprecisiones en el conocimiento de D_l y F_v.

Sin embargo, no debe pensarse que la introducción de un reductor de elevada relación es en general conveniente. Frente a las ventajas aportadas derivadas de la posible simplificación del modelo dinámico del robot, aporta importantes inconvenientes, como son el considerable aumento del rozamiento viscoso y la introducción de holguras y elasticidades en la transmisión.

Como se señaló en el Capítulo 2, dedicado a la morfología del robot, existen en la actualidad algunos modelos de robots denominados de accionamiento directo, en los que se ha prescindido del uso de reductores (**K** = **I**). En ellos se consiguen mayores velocidades y precisiones a costa de una mayor complejidad en el diseño del sistema de control, derivado de no poder considerar τ_p despreciable.

Ejemplo 7.1

Supóngase que el robot de 2 grados de libertad del Ejemplo 5.1 en configuración horizontal, cuyos eslabones están representados en la Figura 7.2, está accionado a través de reductores de factor de reducción 100 (1 vuelta del eje suponen 100 vueltas del motor). Asimismo, supóngase que existe un par de rozamiento viscoso caracterizado por las constantes $f_{v1} = f_{v2} = 5$ Nms/rad.

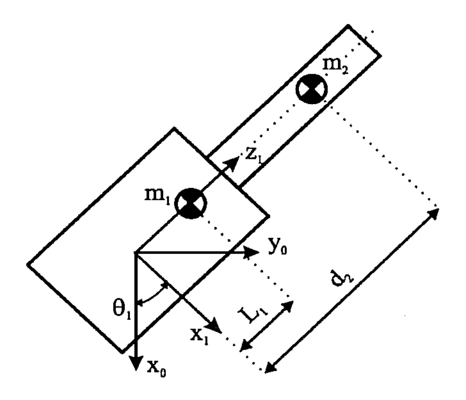


Figura 7.2. Eslabones del robot del Ejemplo 7.1.

Como dimensiones del robot se tomará: $m_1 = 10$ kg , $m_2 = 1$ kg , $L_1 = 0.1$ m , $g = 10$ m/s^2.
Tal y como se obtuvo en el Ejemplo 5.1, el modelo dinámico del robot viene dado por:

$$\mathbf{D} = \begin{bmatrix} m_1 L_1^2 + m_2 d_2^2 & 0 \\ 0 & m_2 \end{bmatrix} = \begin{bmatrix} 0.1 + d_2^2 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{H} = \begin{bmatrix} 2 d_2 m_2 \dot{\theta}_1 \dot{d}_2 \\ -d_2 m_2 \dot{\theta}_1^2 \end{bmatrix} = \begin{bmatrix} 2 d_2 \dot{\theta}_1 \dot{d}_2 \\ -d_2 \dot{\theta}_1^2 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} m_1 g L_1 C_1 + m_2 g d_2 C_1 \\ m_2 g S_1 \end{bmatrix} = \begin{bmatrix} (1 + d_2) 10 C_1 \\ 10 S_1 \end{bmatrix}$$

Por lo tanto se tendrá:

$$\mathbf{K} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}$$

$$\begin{bmatrix} \theta_1 \\ d_2 \end{bmatrix} = \mathbf{K}^{-1} \begin{bmatrix} \theta_{a1} \\ d_{a2} \end{bmatrix} = \begin{bmatrix} 10^{-2}\theta_{a1} \\ 10^{-2} d_{a2} \end{bmatrix}$$

$$\mathbf{D}_1 = \begin{bmatrix} 0.1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{D}_2 = \begin{bmatrix} d_2^2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{K}\,\mathbf{D}_1^{-1}\mathbf{K} = \begin{bmatrix} 10^5 & 0 \\ 0 & 10^4 \end{bmatrix}$$

$$\mathbf{K}^{-1}\mathbf{D}_2\mathbf{K}^{-1} = \begin{bmatrix} 10^{-4} d_2^2 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 10^{-8} d_{a2}^2 & 0 \\ 0 & 0 \end{bmatrix}$$

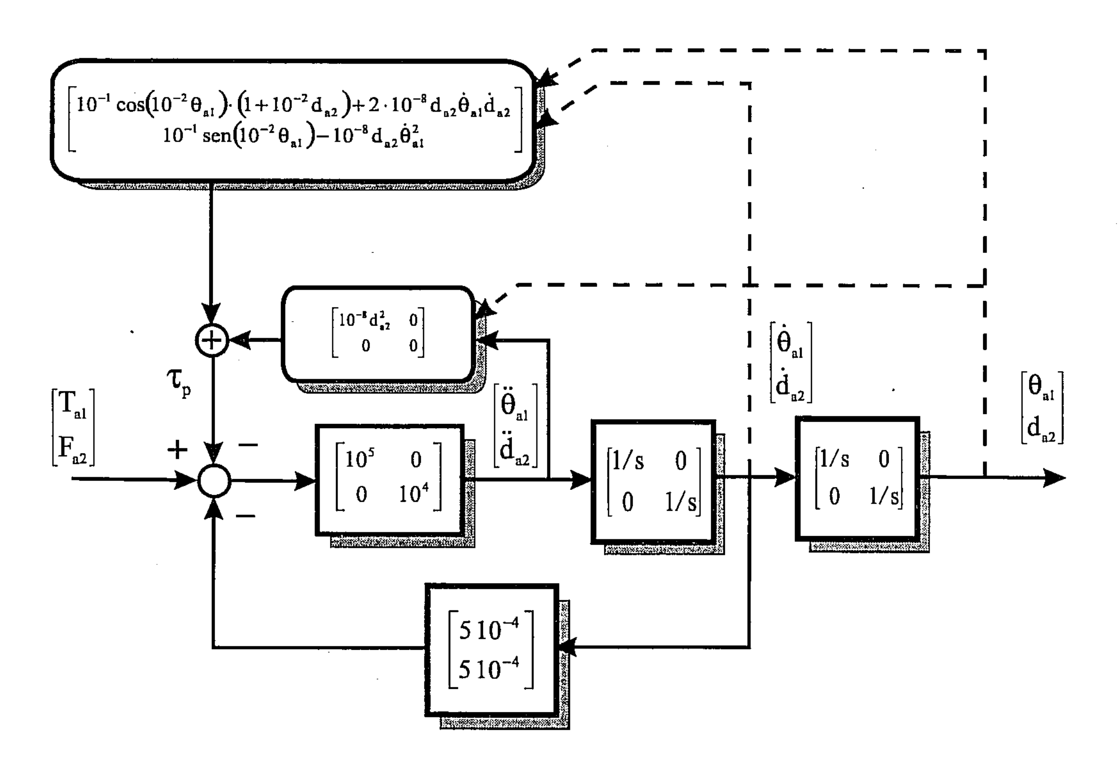


Figura 7.3. Diagrama de bloques del modelo dinámico del robot del Ejemplo 7.1.

$$\mathbf{K}^{-1}\mathbf{H} = \begin{bmatrix} 2 \cdot 10^{-2} d_2 \dot{\theta}_1 \dot{d}_2 \\ -10^2 d_2 \dot{\theta}_1^2 \end{bmatrix} = \begin{bmatrix} 2 \cdot 10^{-8} d_{a2} \dot{\theta}_{a1} \dot{d}_{a2} \\ -10^{-8} d_{a2} \dot{\theta}_{a1}^2 \end{bmatrix}$$

$$\mathbf{K}^{-1}\mathbf{C} = \begin{bmatrix} 10^{-1} C_1 (1 + d_2) \\ 10^{-1} S_1 \end{bmatrix} = \begin{bmatrix} 10^{-1} \cos(10^{-2}\theta_{a1})(1 + 10^{-2} d_{a2}) \\ 10^{-1} \mathrm{sen}(10^{-2}\theta_{a1}) \end{bmatrix}$$

$$\mathbf{F}_{va} = \mathbf{K}^{-1}\mathbf{F}_v\mathbf{K}^{-1} = \begin{bmatrix} 10^{-4} f_{v1} \\ 10^{-4} f_{v2} \end{bmatrix} = \begin{bmatrix} 5\,10^{-4} \\ 5\,10^{-4} \end{bmatrix}$$

Con lo que la representación en forma de diagrama de bloques del modelo dinámico del robot quedará como en la Figura 7.3. Puede apreciarse cómo la aportación de par τ_p es significativamente pequeña, pudiendo ser despreciada.

7.1.2. Control PID

En casos como los expuestos en el epígrafe anterior, el control del robot puede ser tratado como el control de n ejes independientes, incluyendo en el par perturbador τ_p el efecto de los parámetros no constantes de la diagonal de la matriz **D**, así como sus elementos no diagonales y los pares generados por la gravedad y movimientos relativos (matrices **C** y **H**). En estos casos y suponiendo el accionamiento de tipo eléctrico, se tendrá (ver epígrafe 5.6.1) que el correspondiente diagrama de bloques de cada eje sería el mostrado en la Figura 7.4. Por claridad se prescinde de aquí en adelante del factor de reducción aplicado en el eje.

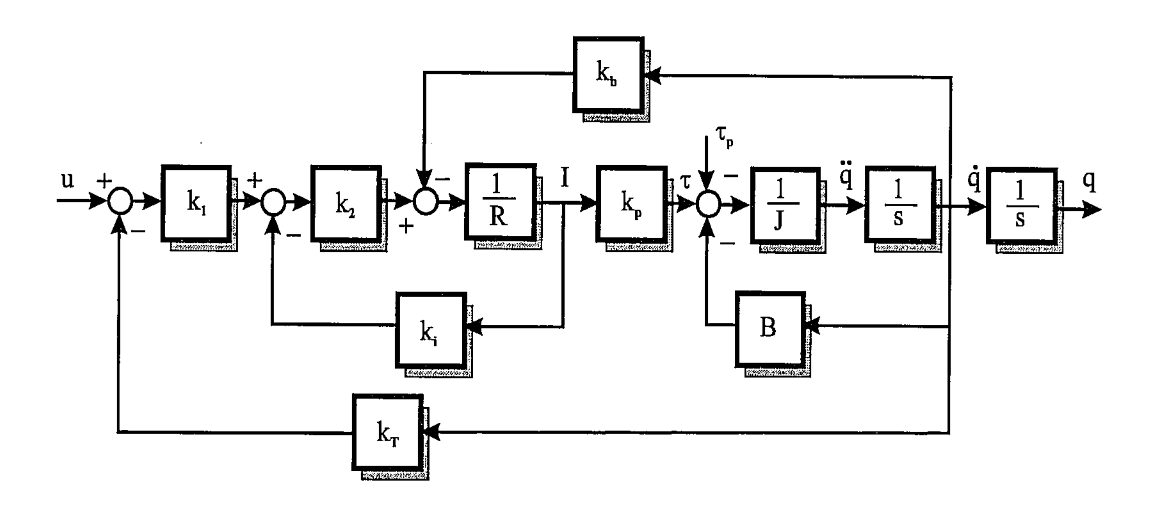


Figura 7.4. Diagrama de bloques de un accionamiento eléctrico.

Los términos J y B son, respectivamente, inercia y rozamiento viscoso vistos por el actuador, e incluyen tanto los efectos del eslabón y la articulación como los del motor y la transmisión.

En la práctica, la calidad de los sistemas de accionamiento es tal que se consigue una relación lineal entre la tensión de mando y el par proporcionado por el accionamiento, de modo que:

$$\tau = k\,u \qquad [7.7]$$

con lo que al ser:

$$\tau = (Js + B)sq + \tau_p \qquad [7.8]$$

resulta:

$$q = \frac{k}{(Js + B)s}\left(u - \frac{1}{k}\tau_p\right) \qquad [7.9]$$

Suponiendo τ_p inexistente y perfectamente conocidos k, J y B, podría plantearse obtener por inversión del modelo la tensión de mando u(t) que resultaría en una trayectoria q(t) preestablecida. Sin embargo, en la práctica ninguna de las dos premisas planteadas es cierta, por lo que es habitual utilizar un esquema de control realimentado como el indicado en la Figura 7.5.

En este esquema, R(s) representa un regulador tipo PID de función de transferencia:

$$R(s) = k_p + \frac{k_i}{s} + k_d s = \frac{1}{s}\left(k_p s + k_i + k_d s^2\right) \qquad [7.10]$$

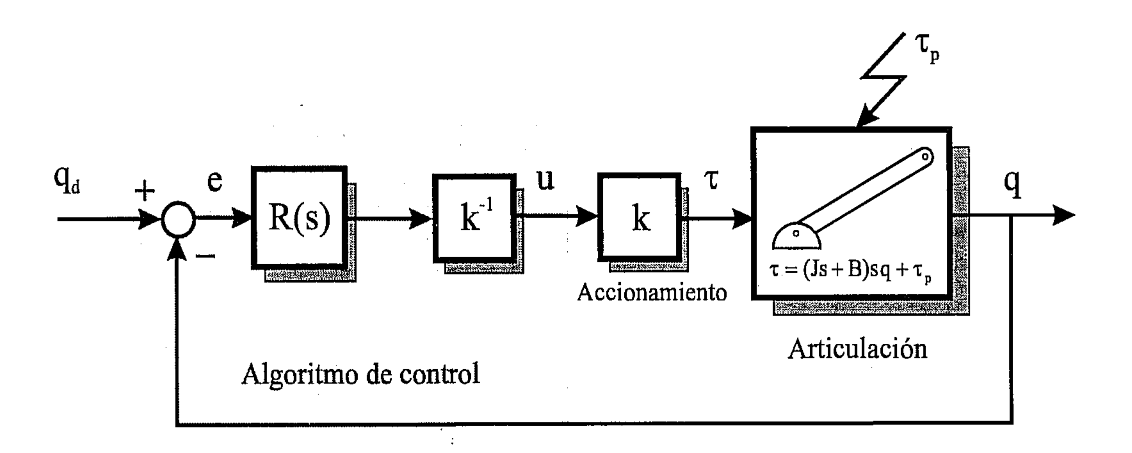


Figura 7.5. Esquema de algoritmo de control basado en PID.

El término k^{-1} se ha introducido al objeto de que la salida del regulador R(s) sea directamente el par τ de entrada a la articulación. La señal de mando u (ley de control) en los accionamientos será entonces:

$$u = \frac{1}{s}\left(k_p s + k_i + k_d s^2\right)\frac{1}{k} e \qquad [7.\ 11]$$

Esta expresión representa la ley de control que permite calcular cuál es el par que se pretende que den los actuadores del robot. A la vista del diagrama de bloques, se tiene que:

$$\tau = R(s)e \qquad [7.\ 12]$$

y haciendo uso de la ecuación [7.8] que establece el comportamiento dinámico de la articulación, se tendrá:

$$\begin{aligned} &(Js + B)sq + \tau_p = R(s)e \quad \Rightarrow \\ &\tau_p = R(s)e - (Js + B)sq \end{aligned} \qquad [7.\ 13]$$

Considerando que $e = q_d - q$ y sumando $(Js + B)sq_d$ en ambos miembros de [7.13] se tiene:

$$\tau_p + (Js + B)sq_d = \left[R(s) + (Js + B)s\right]e \qquad [7.\ 14]$$

Con lo que el error entre la trayectoria deseada $q_d(t)$ y la real q(t) vendrá dada por:

$$e(s) = \frac{\tau_p + (Js + B)sq_d}{R(s) + (Js + B)s} \qquad [7.\ 15]$$

Esta expresión de e(s) indica que si no se cambia la posición deseada ($q_d(s) = 0$), pero aparece una perturbación en forma de escalón ($\tau_p = 1/s$), el error valdrá:

$$e(s) = \frac{1}{\left(k_p s + k_i + k_d s^2\right) + (Js + B)s^2} \qquad [7.\ 16]$$

Supuesto el sistema estable, el valor final del error podrá obtenerse por aplicación del teorema del valor final:

$$\Delta e(\infty) = \lim_{s \to 0} s \cdot e(s) = 0 \qquad \text{siempre que } k_i \neq 0 \qquad [7.\ 17]$$

Obsérvese que de utilizar un regulador sin acción integral (regulador PD, $k_i = 0$), el error ante perturbación escalón valdría:

$$e(s) = \frac{1}{s\left[\left(k_p + k_d s\right) + (Js + B)s\right]} \quad [7.18]$$

siendo su valor final:

$$\Delta e(\infty) = \lim_{s \to 0} s \cdot e(s) = \frac{1}{k_p} \quad [7.19]$$

que, aunque disminuirá al aumentar la ganancia proporcional k_p, no llegará a hacerse cero.

Por lo tanto, el esquema de control propuesto, utilizando un regulador PID (con acción integral no nula) garantiza, siempre que el diseño del regulador resulte en un sistema estable, que el efecto de los pares perturbadores en forma de escalón se anula.

Por otra parte, suponiendo que no existiese perturbación, pero se demandase a la articulación que evolucionase según una determinada trayectoria $q_d(t)$, se tendría, según la ecuación [7.15]:

$$e(s) = \frac{(Js + B)sq_d}{R(s) + (Js + B)s} \quad [7.20]$$

con lo que al ser por definición $e = q_d - q$ se tendrá:

$$\left[(Js + B)s + R(s) - (Js + B)s\right]q_d = \left[(Js + B)s + R(s)\right]q \quad \Rightarrow$$

$$\frac{q}{q_d} = \frac{R(s)}{R(s) + (Js + B)s} = 1 - \frac{(Js + B)s}{R(s) + (Js + B)s} \quad [7.21]$$

Es decir, $q(t)$ no coincide a lo largo del tiempo con $q_d(t)$. Así, ante una petición de cambio instantáneo en la posición deseada de la articulación ($q_d(s) = 1/s$) se tendría:

$$q(s) = \frac{1}{s}\frac{R(s)}{R(s) + (Js + B)s} = \frac{\frac{1}{s}\left(k_p + \frac{k_i}{s} + k_d s\right)}{\left(k_p + \frac{k_i}{s} + k_d s\right) + (Js + B)s} \quad [7.22]$$

Resultando que $q(t)$ y $q_d(t)$ no coinciden salvo, suponiendo el sistema estable, en el valor final ($t \to \infty$) en el que q vale 1, coincidiendo con la entrada.

En definitiva, el control PID permite anular el efecto de una posible perturbación en escalón en régimen permanente, así como asegurar que finalmente q y q_d coinciden, sin embargo,no consigue que q siga la evolución de q_d a lo largo del tiempo.

7.1.3. Control PID con prealimentación

Como se ha visto, la utilización del control PID permite anular el efecto de la perturbación en régimen permanente, así como conseguir que la salida q(t) alcance a la entrada pasado el suficiente tiempo. Sin embargo, sería deseable que ante una trayectoria de entrada $q_d(t)$, la salida coincidiese con ella en todo momento, consiguiendo no sólo un error nulo en regimen permanente, sino también un error de seguimiento nulo.

Para conseguir tal efecto puede utilizarse un control PID con prealimentación, basado en un preciso conocimiento del modelo de la articulación (J y B). La Figura 7.6 representa el esquema de control PID con prealimentación, donde a la señal de mando u se le ha incorporado una prealimentación de velocidad y de aceleración, siendo ahora la ley de control:

$$\begin{aligned} &u = k^{-1}\left[R(s)e + (Js + B)sq_d\right] \\ &\text{con} \quad R(s) = k_p + \frac{k_i}{s} + k_d s \end{aligned} \qquad [7.\ 23]$$

Haciendo uso de la característica tensión-par del actuador [7.7] y del modelo dinámico de la articulación [7.8], se tendrá:

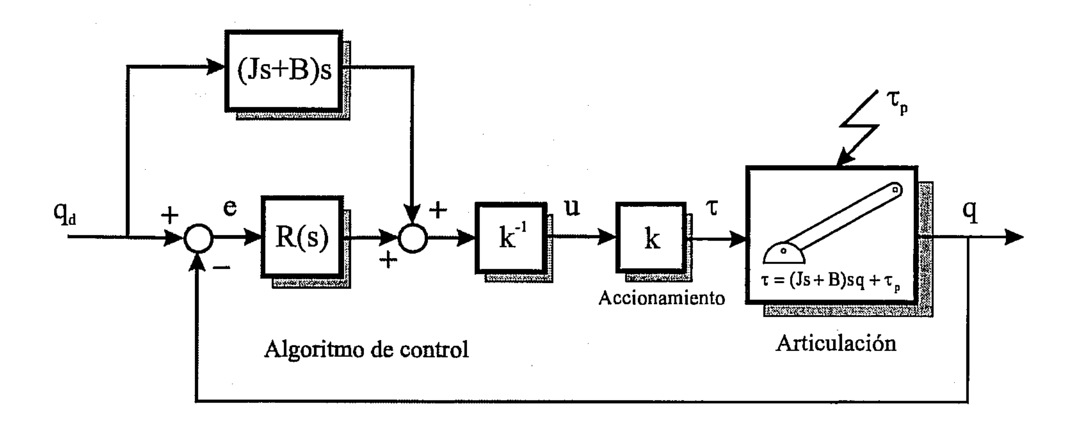


Figura 7.6. Esquema de algoritmo de control basado en PID con prealimentación.

$$\left.\begin{array}{l}\tau = k\,u = R(s)\,e + (Js + B)s\,q_d \\ \tau = (Js + B)s\,q + \tau_p\end{array}\right\} \Rightarrow$$

$$R(s)\,e + (Js + B)s\,q_d = (Js + B)s\,q + \tau_p \quad \Rightarrow$$

$$\big(R(s) + (Js + B)s\big)e = \tau_p \quad \Rightarrow$$

$$e = \frac{\tau_p}{R(s) + (Js + B)s} \qquad [7.24]$$

Como se observa, a diferencia de la expresión [7.15], obtenida para el error cuando sólo se utiliza control PID, el error obtenido con el uso de control PID más una prealimentación de aceleración y velocidad, no depende de q_d. Luego, si bien ante presencia de perturbación el error se comportará como en el caso anterior (tendiendo a cero cuando τ_p es un escalón), ante una entrada en q_d, no tendrá valor alguno, consiguiéndose un seguimiento absoluto de la trayectoria deseada, es decir, $q(t) = q_d(t)$ para toda la trayectoria.

Debe advertirse una vez más, ante el posible optimismo de los resultados anteriores, que el seguimiento absoluto de la trayectoria deseada $q_d(t)$ ante ausencia de perturbación se conseguiría solo si J, B y k fuesen perfectamente conocidas. En la práctica, habrá que conformarse con tener una buena estimación de estos valores y admitir ciertas desviaciones entre q y q_d.

7.1.4. Control PD con compensación de gravedad

Como se vio en 7.1.2, el control PID asegura que ante perturbación se anula el error en régimen permanente. Sin embargo, de usar sólo un regulador PD no se consigue este efecto. Por otra parte, es sabido que la inclusión de la acción integral (k_i) necesaria para eliminar el error en régimen permanente, tiende a empeorar la respuesta dinámica de la articulación (además de tener otros inconvenientes, como se verá en 7.4.2). También en 7.1.3 se ha visto cómo la prealimentación puede ser utilizada para eliminar el error de seguimiento, siempre que se tenga un buen modelo del sistema.

Recordando el esquema presentado en la Figura 7.1, se observa que en régimen permanente el término τ_p depende sólo de la gravedad (matriz **C**(q)) y eventualmente de perturbaciones aleatorias, puesto que $\dot{q}$ y $\ddot{q}$ son nulas en esta situación.

El control PD con compensación de gravedad saca partido de estos hechos, no utilizando la acción integral para anular el error permanente, sino una aportación G(q) que compensa el valor τ_p en régimen permanente. Al ser el valor G(q) debido tan solo al término de gravedad, su conocimiento y computo en tiempo real es relativamente simple. La Figura 7.7 recoge el esquema de control correspondiente, en el que ahora R(s) es un control tipo PD, $R(s) = k_p + k_d s$, resultando la ley de control:

$$u = k^{-1}\big[R(s)\,e + G(q)\big] \qquad [7.25]$$

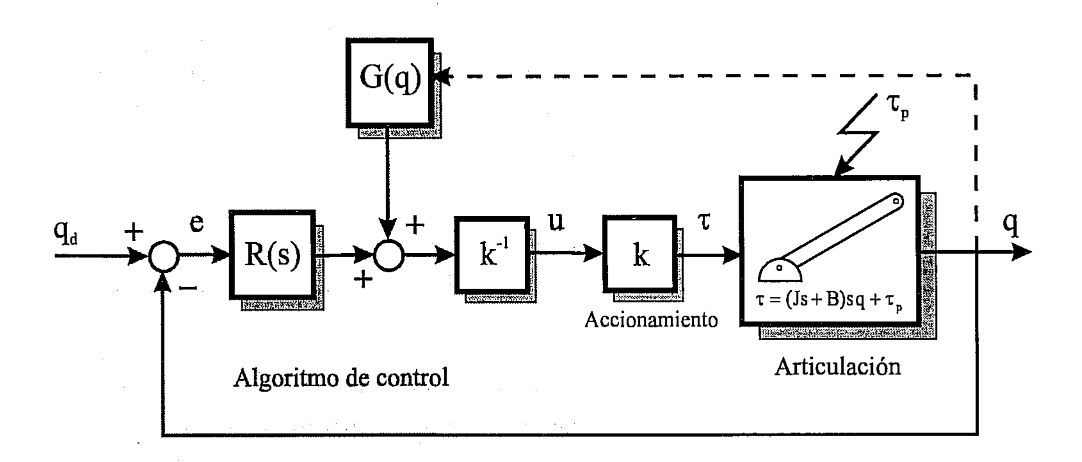


Figura 7.7. Esquema del algoritmo de control basado en PD con compensación de gravedad.

con lo que al considerar la relación tensión-par del actuador [7.7] y el modelo dinámico de la articulación [7.8], se tiene:

$$\left.\begin{aligned} \tau &= k\,u = R(s)e + G(q) \\ \tau &= (Js + B)sq + \tau_p \end{aligned}\right\} \Rightarrow$$

$$R(s)e + G(q) = (Js + B)sq + \tau_p \quad \Rightarrow$$

$$\big(R(s) + (Js + B)s\big)e = (Js + B)sq_d + \tau_p - G(q) \quad \Rightarrow$$

$$e = \frac{(Js + B)sq_d + \tau_p - G(s)}{R(s) + (Js + B)s} \qquad [7.26]$$

Suponiendo que el robot se encuentra parado, esto es, que $\Delta q_d = 0$, se tendrá:

$$\Delta e(\infty) = \lim_{s \to 0} s \frac{\tau_p - G(q)}{R(s) + (Js + B)s} \qquad [7.27]$$

Si en régimen permanente τ_p viene dado sólo por los efectos de la gravedad $\tau_p = C$, se tendrá que para que el error permanente sea nulo:

$$G(q) = \tau_p = C(q) \qquad [7.28]$$

Por lo tanto, una compensación calculada a partir de los términos de gravedad, permitirá utilizar un controlador PD asegurando que el error permanente es nulo.

7.2. CONTROL MULTIARTICULAR

La consideración de que cada articulación no se ve afectada más que por sus propias características dinámicas, siendo el efecto de las demás (representado por el par perturbador τ_p) poco significativo, no es siempre aceptable.

Frente a las técnicas basadas en el supuesto desacoplamiento de las articulaciones, las técnicas de control acoplado contemplan el modelo dinámico real del robot, haciendo uso del conocimiento del mismo para tratar de desacoplar el sistema.

7.2.1. Desacoplamiento por inversión del modelo

Considerando una vez más el modelo dinámico del robot, incluyendo esta vez un posible par exterior τ_e:

$$\tau = \mathbf{D}(q)\ddot{\mathbf{q}} + \mathbf{H}(q,\dot{q}) + \mathbf{C}(q) + \mathbf{F}_v\dot{\mathbf{q}} + \tau_e = \mathbf{D}(q)\ddot{\mathbf{q}} + \mathrm{N}(q,\dot{q}) + \tau_e \qquad [7.29]$$

donde en $\mathrm{N}(q,\dot{q})$ se han incluido todos los pares distintos del de inercia (Coriolis, gravedad, rozamiento, etc.).

Para desacoplar este modelo puede recurrirse al esquema representado en la Figura 7.8, en el que la tensión de mando de los actuadores **u** se obtiene a partir de la señal $\hat{\tau}$ y de las matrices $\mathbf{D}(q)$ y $\mathrm{N}(q,\dot{q})$, siendo:

$$\left.\begin{aligned} \tau &= \mathbf{K}\,\mathbf{u} = \left[\mathbf{D}(q)\hat{\tau} + \mathrm{N}(q,\dot{q})\right] \\ \tau &= \mathbf{D}(q)\ddot{\mathbf{q}} + \mathrm{N}(q,\dot{q}) + \tau_e \end{aligned}\right\} \Rightarrow$$

$$\hat{\tau} = \ddot{\mathbf{q}} + \mathbf{D}^{-1}(q)\tau_e \qquad [7.30]$$

Resultando que el esquema de la Figura 7.8-a equivale al de la Figura 7.8-b, en el que se observa que en ausencia de par externo τ_e, la relación entre $\hat{\tau}$ y **q** es desacoplada y lineal.

Una vez desacoplado el sistema, es sencillo calcular el par $\hat{\tau}$ necesario para seguir la trayectoria deseada $\mathbf{q}_d$. Bastará para ello con que $\hat{\tau} = \ddot{\mathbf{q}}_d = s^2\mathbf{I}\mathbf{q}_d$, con lo que:

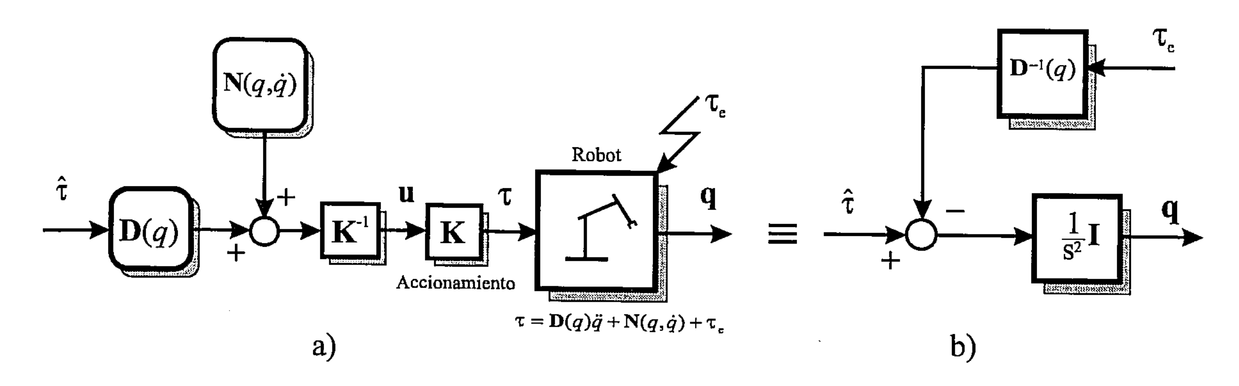


Figura 7.8. Desacoplamiento por inversión del modelo.

$$\ddot{\mathbf{q}}_d = \ddot{\mathbf{q}} + \mathbf{D}^{-1}(q)\tau_e \qquad [7.\ 31]$$

Como se observa, en ausencia de par exterior τ_e si $\hat{\tau} = \ddot{\mathbf{q}}_d$, la trayectoria deseada $\mathbf{q}_d$ y la real coinciden. Por lo tanto, la ley de control correspondiente al método de desacoplamiento del modelo supone que la tensión de mando a proporcionar a los accionamientos sea (Figura 7.9):

$$\mathbf{u} = \mathbf{K}^{-1}\left[\mathbf{D}(q)\,s^2\mathbf{I}\mathbf{q}_d + \mathbf{N}(q,\dot{q})\right] \qquad [7.\ 32]$$

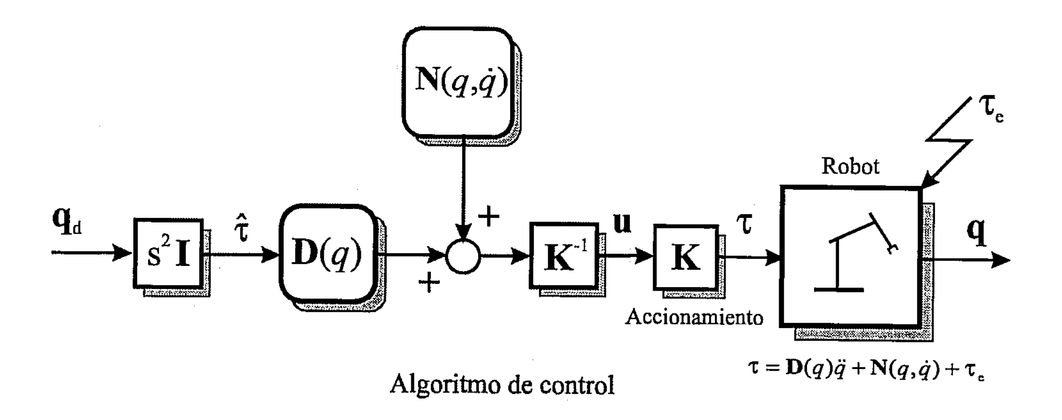


Figura 7.9. Esquema de algoritmo de control basado en desacoplamiento por inversión del modelo.

7.2.2. Control PID con prealimentación

Una vez conseguido independizar dinámicamente el movimiento de las articulaciones del robot, cabe aplicar cualquiera de las técnicas de control presentadas en 7.1 para el caso de ejes independientes. En concreto, la presencia de perturbaciones τ_e aleatorias no modeladas, e incluso las discrepancias entre los valores reales y los estimados en las matrices **D** y **N**, pueden ser en cierta medida compensados con el clásico esquema de control PID.

En el diagrama de bloques de la Figura 7.10, todas las variables son vectores de dimensión n y el bloque **R**(s) es una matriz diagonal (n x n) de reguladores tipo PID:

$$\mathbf{R}(s) = \begin{bmatrix} k_{p1} + \frac{k_{i1}}{s} + k_{d1}s & 0 & \cdots & 0 \\ 0 & k_{p2} + \frac{k_{i2}}{s} + k_{d2}s & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & k_{pn} + \frac{k_{in}}{s} + k_{dn}s \end{bmatrix} \quad [7.33]$$

Este diagrama es equivalente al de la Figura 7.9 añadiendo un bucle PID, resultando similar, tal y como se ha justificado en 7.2.1, al de la Figura 7.11. En él se verifica:

$$\left.\begin{array}{l} \hat{\tau} = s^2 \mathbf{I} \mathbf{q}_d + \mathbf{R}(s)\mathbf{e} \\ \tau = \mathbf{D}(q)\hat{\tau} + \mathbf{N}(q,\dot{q}) \end{array}\right\} \Rightarrow \tau = \mathbf{D}(q)\left[s^2 \mathbf{I} \mathbf{q}_d + \mathbf{R}(s)\mathbf{e}\right] + \mathbf{N}(q,\dot{q}) \quad [7.34]$$

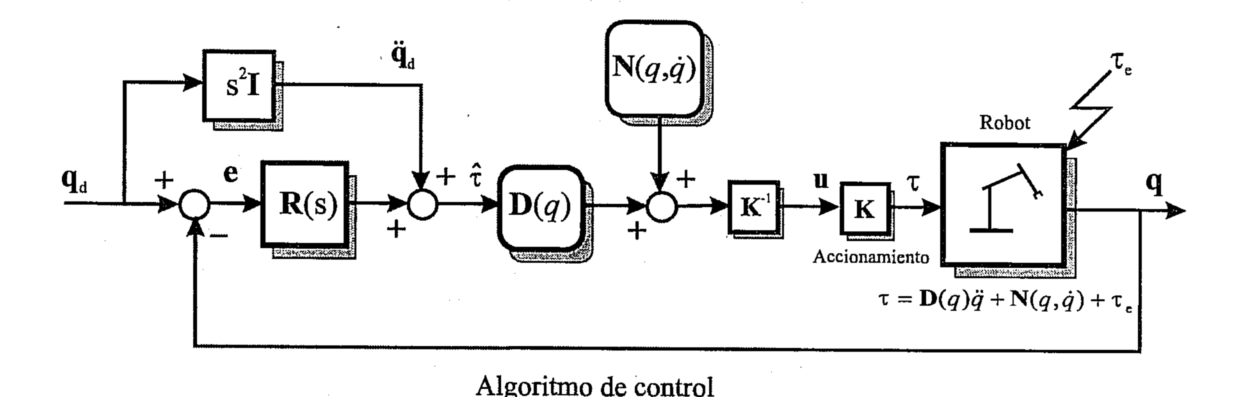


Figura 7.10. Esquema de algoritmo de control basado en PID y prealimentación sobre el sistema desacoplado.

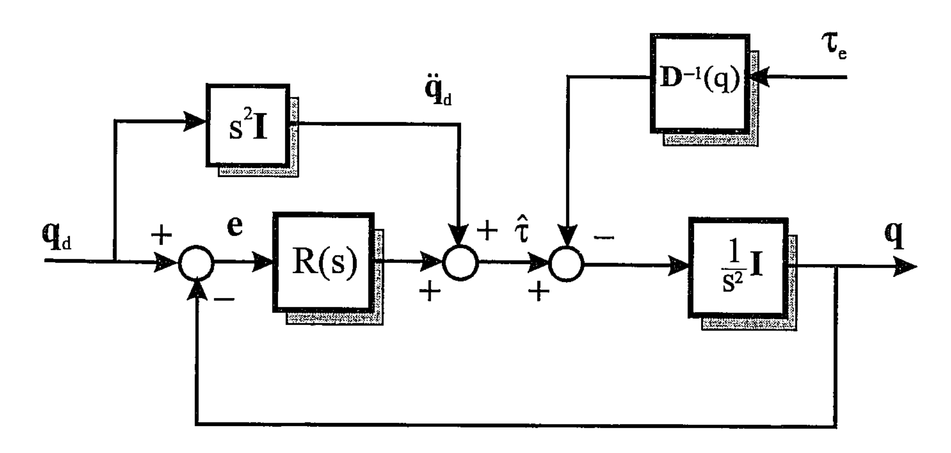


Figura 7.11. Esquema de algoritmo de control basado en PID y prealimentación sobre el sistema desacoplado.

Por otra parte, el modelo dinámico del robot [7.29], establece que:

$$\tau = \mathbf{D}(q)\,s^2\mathbf{I}\mathbf{q} + \mathbf{N}(q,\dot{q}) + \tau_e \qquad [7.\ 35]$$

Igualando las ecuaciones [7.34] y [7.35] se obtiene:

$$\mathbf{D}(q)\left[s^2\mathbf{I}\mathbf{q}_d + \mathbf{R}(s)\,\mathbf{e}\right] = \mathbf{D}(q)s^2\mathbf{I}\mathbf{q} + \tau_e \quad \Rightarrow$$
$$\mathbf{D}(q)\left[s^2\mathbf{I} + \mathbf{R}(s)\right]\mathbf{e} = \tau_e \quad \Rightarrow$$

$$\mathbf{e} = \left[\mathbf{D}(q)\left(s^2\mathbf{I} + \mathbf{R}(s)\right)\right]^{-1}\tau_e \qquad [7.\ 36]$$

Expresión que pone de manifiesto que la utilización de reguladores PID y prealimentación de aceleración independiente por cada eje, junto con el desacoplamiento por la inversión del modelo da lugar a que:

1. En ausencia de perturbaciones ($\tau_e = 0$) el error es siempre nulo y, por tanto, $\mathbf{q} = \ddot{\mathbf{q}}_d$.
2. Ante una perturbación en forma de escalón ($\tau_e = 1/s$), el error en régimen permanente se anula.

$$\mathbf{e}(\infty) = \lim_{s\to 0} s\left[s^2\mathbf{I} + \mathbf{R}(s)\right]^{-1}\mathbf{D}^{-1}(q)\frac{1}{s} = \lim_{s\to 0}\mathbf{R}(s)^{-1}\mathbf{D}^{-1}(q) = 0 \qquad [7.\ 37]$$

puesto que:

$$\mathbf{R}(s)^{-1} = s\begin{bmatrix} \dfrac{1}{k_{p1}s + k_{i1} + k_{d1}s^2} & 0 & \cdots & 0 \\ 0 & \dfrac{1}{k_{p2}s + k_{i2} + k_{d2}s^2} & \cdots & 0 \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \dfrac{1}{k_{pn}s + k_{in} + k_{dn}s^2} \end{bmatrix} \qquad [7.38]$$

De la expresión [7.34] y considerando la relación proporcional entre la tensión de mando del motor y el par proporcionado por éste, se tiene:

$$\mathbf{u} = \mathbf{K}^{-1}\tau = \mathbf{K}^{-1}\mathbf{D}(q)\left[s^2\mathbf{I}\mathbf{q}_d + \mathbf{R}(s)\mathbf{e}\right] + \mathbf{K}^{-1}\mathbf{N}(q,\dot{q}) \qquad [7.39]$$

siendo ésta la denominada ley de control que establece el cálculo del par que debe realizar la unidad de control para fijar las consignas de par de los actuadores.

La evaluación en tiempo real de la expresión [7.39] puede ser excesivamente costosa. En ocasiones, y considerando que las velocidades $\dot{\mathbf{q}}$ de las articulaciones no son muy elevadas (del orden de media vuelta por segundo) puede simplificarse su cálculo despreciando el efecto de la matriz $\mathbf{H}(q,\dot{q})$ y de la matriz $\mathbf{F}_v(q)\dot{\mathbf{q}}$ sobre el término $\mathbf{N}(q,\dot{q})$ que entonces coincidirá con la matriz de gravedad $\mathbf{C}(q)$.

La expresión [7.39] es el caso general de control por **par calculado** y según se consideren o no ciertos términos, se tendrán diferentes casos particulares.

7.3. CONTROL ADAPTATIVO

Como se ha visto en el epígrafe anterior, si se conoce perfectamente el modelo dinámico del robot, es posible utilizar la técnica de desacoplamiento por inversión del modelo para, en ausencia de perturbación, conseguir que la trayectoria real coincida exactamente con la deseada.

Sin embargo, en la práctica esta técnica tiene una aplicación limitada, bien porque los parámetros que definen el modelo (inercias, rozamientos, etc.) no son conocidos con precisión, o bien porque el cálculo continuo en tiempo real del modelo inverso es excesivamente lento.

La técnica del **control adaptativo** se puede aplicar con buenos resultados en aquellas ocasiones en las que el modelo, aun siendo conocido, cambia continuamente por cambiar las condiciones de funcionamiento. La idea básica del control adaptativo es modificar en tiempo real los parámetros que

definen el regulador (PID por ejemplo) de acuerdo al comportamiento instántaneo del sistema [RODRIGUEZ-96].

En el caso de un robot, es evidente que el comportamiento del sistema, y por lo tanto el modelo del mismo, cambia con los valores de sus variables articulares y con la carga que transporta. De este modo, supuesto que se hubiese ajustado adecuadamente un controlador para unas condiciones determinadas de localización del robot y carga manipulada, este ajuste no sería válido cuando alguna de estas condiciones cambiase.

Muchos robots industriales tienen un único controlador ajustado para unas condiciones intermedias de funcionamiento y asegurando que los movimientos del robot se mantienen estables en otras condiciones, aunque con su calidad disminuida (tiempo de establecimiento, oscilación, precisión, etc.). La utilización de la técnica de control adaptativo permite reajustar los parámetros del regulador cada vez que el comportamiento del robot se modifica, de modo que los movimientos de éste se aproximen siempre a las especificaciones deseadas.

Existen diferentes esquemas de control que entran dentro del concepto de control adaptativo. En todos ellos se utilizan procedimientos de identificación de los parámetros del modelo del sistema y algoritmos de cálculo de los parámetros del regulador.

El control por planificación de ganancias puede ser considerado como un caso simple de control adaptativo, en el que un número determinado de reguladores están precalculados para diferentes condiciones de funcionamiento. Según se detecta que éstas se modifican se conmuta de un regulador a otro.

El control adaptativo con modelo de referencia realiza una comparación en línea entre el comportamiento del sistema real y el deseado. El error entre ambos comportamientos se utiliza para modificar los parámetros del regulador.

El control adaptativo autoajustable identifica continuamente el modelo del sistema y utiliza algún algoritmo de diseño para ajustar los parámetros del regulador. Como caso particular de éste se comentará el control de par calculado adaptativo.

7.3.1. Control adaptativo por planificación de ganancias (GS)

En el método de planificación de ganancias, una vez seleccionado el tipo de control a utilizar (por ejemplo PID con prealimentación de aceleración independiente por cada eje) se ajustan los parámetros del mismo para un número determinado de condiciones de funcionamiento.

Las variables que hacen que estas condiciones de funcionamiento se modifiquen son la carga que manipula el robot y los valores de las coordenadas articulares. Durante el funcionamiento, los valores de las coordenadas articulares son conocidas por la unidad de control del robot (ya sean las deseadas, generadas por el control cinemático, o las reales, obtenidas por los captadores de posición). Asimismo, la carga puede ser conocida, bien porque el programa del robot incluya instrucciones que permitan la especificación de la misma, bien porque el extremo del robot cuente con un sensor adecuado o bien porque se incluyan algoritmos que permitan calcular la carga a través del comportamiento de magnitudes físicas del sistema (intensidad de accionamientos eléctricos, presión diferencial en hidráulicos, etc.) [PAUL-81].

Con estos datos, el algoritmo de planificación de ganancias hace uso de una tabla en la que están especificados los parámetros de los reguladores a utilizar para cada uno de los grados de libertad. En

el caso de que el punto de funcionamiento no coincida exactamente con las entradas a la tabla, puede utilizarse alguna técnica de interpolación.

Debe considerarse que en el caso general de un robot de 6 grados de libertad se tendría hasta 7 variables a ir modificando para establecer las diferentes condiciones de trabajo. Esto puede dar lugar a un número excesivo de puntos de funcionamiento y, por lo tanto, de reguladores distintos. Así, suponiendo que cada grado de libertad tuviese un recorrido de 180°, discretizados en intervalos de 15°, y que la carga variase de 0 a 50 kg, discretizándose en intervalos de 5 kg, se tendría que el número de puntos de funcionamiento distintos a considerar sería:

$$\frac{50}{5} \cdot \left(\frac{180}{15}\right)^6 = 2.985894 \cdot 10^7$$

A la vista de este resultado se concluye que no es posible realizar la citada discretización. Para disminuir el número de modelos diferentes puede comenzarse por hacer las siguientes consideraciones:

1. Los movimientos de los últimos 3 grados de libertad, destinados a orientar el extremo del robot, suponen pequeñas variaciones en la dinámica del robot, por cuanto las longitudes de las correspondientes articulaciones (parámetro d en los parámetros D-H) son reducidas.
2. En muchos robots, como es el caso de los articulares, el valor del primer grado de la libertad está en un plano perpendicular a los demás. Eso origina que el efecto de su movimiento sobre los demás sea mínimo, especialmente si las velocidades de movimiento no son elevadas.

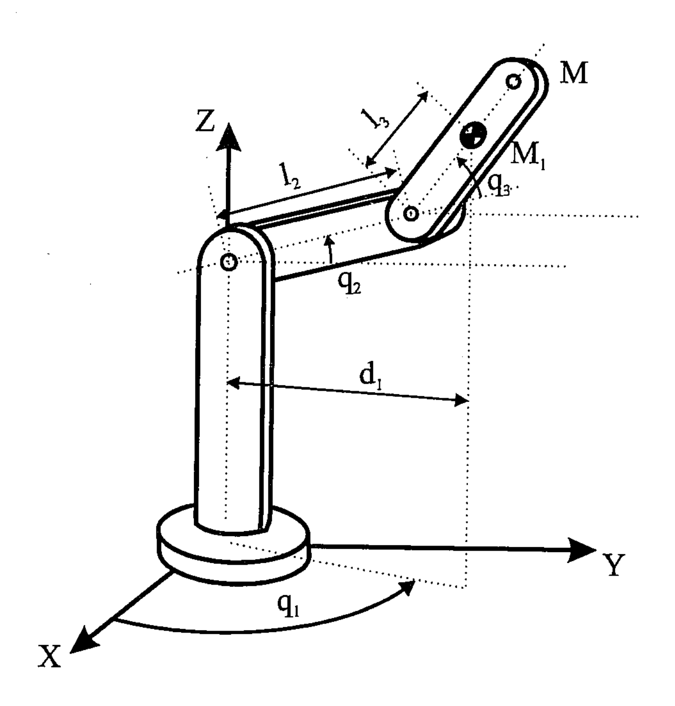


Figura 7.12. Robot articular de 3 grados de libertad.

Además, un adecuado estudio del robot permite combinar los ejes para disminuir el número de entradas a la tabla. Así, en el robot articular de la Figura 7.12, la dinámica del eje 1 se ve afectada por una función de la masa reducida M_l (de los eslabones 2, 3 y masa manipulada M) y de la distancia d_l de la masa reducida al eje de giro del grado de libertad 1, cuyo valor es:

$$d_1 = l_2 \cos q_2 + l_3 \cos(q_2 + q_3)$$

reduciéndose así el número de entradas a la tabla.

Del mismo modo, el eje 3 por ejemplo, vería afectada su dinámica por una función de la masa reducida M_3 (del eslabón 3 y masa manipulada M) y de la distancia d_2 de ésta al eje de giro del grado de libertad, reduciéndose del mismo modo el número de entradas a la tabla. (Consideraciones similares pueden hacerse sobre los demás ejes.)

Un segundo problema a considerar en la utilización de un control por planificación de ganancias es, que si bien para los puntos de funcionamiento deseado se tiene asegurado la estabilidad y buen funcionamiento del sistema (con estas premisas se ha diseñado el regulador), para puntos intermedios, distintos de los considerados, no está garantizada ni siquiera la estabilidad.

Este problema puede ser tratado de manera teórica con técnicas similares a las utilizadas en control robusto [ASTRÖM-95], o bien de manera más práctica realizando un número amplio de pruebas en diferentes condiciones de funcionamiento.

La Figura 7.13 representa el esquema de funcionamiento de un control monoarticular con algoritmo PID y prealimentación de aceleración usando una planificación de ganancias. Téngase en cuenta que la determinación de los parámetros k_a, k_p, k_i, k_d debe realizarse para cada uno de los grados de libertad sobre los que se desee reajustar el regulador.

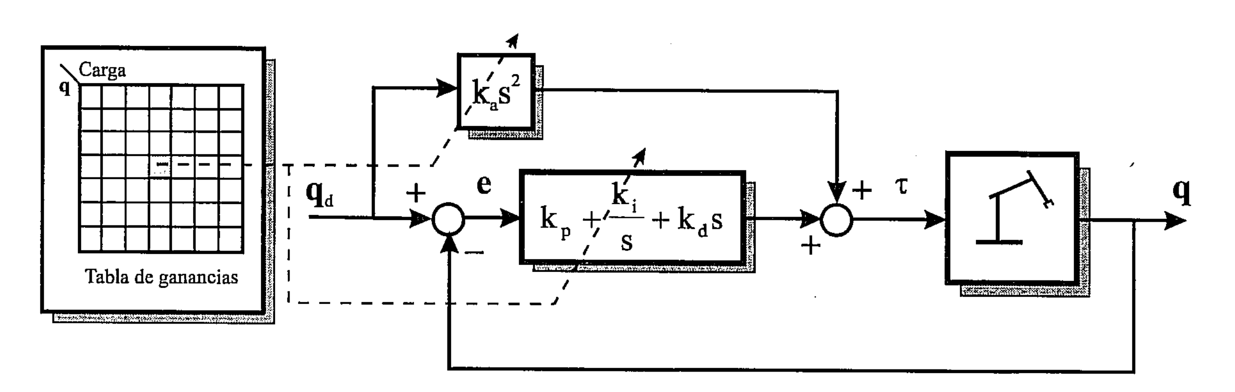


Figura 7.13. Control monoarticular con planificación de ganancias.

7.3.2. Control adaptativo con modelo de referencia (MRAC)

En este tipo de control adaptativo se parte de un comportamiento ideal del sistema, definido por un modelo de referencia (la elección del mismo no es trivial). La diferencia instantánea entre el

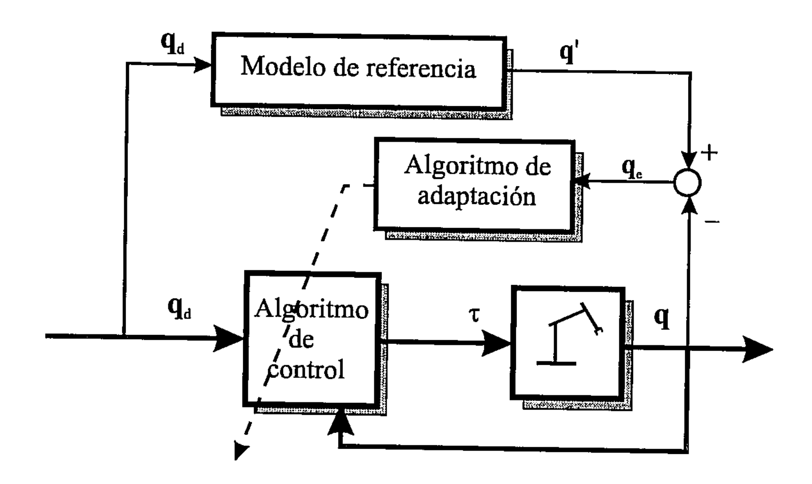


Figura 7.14. Funcionamiento de un controlador MRAC.

comportamiento real del robot **q** y el modelo deseado **q'** se utiliza para calcular los parámetros del controlador utilizado.

La Figura 7.14 representa el funcionamiento de un controlador MRAC. Junto con el bucle de control normal, dibujado en trazo grueso, aparecen otros bloques que componen el control adaptativo. El algoritmo de control puede ser cualquiera de los presentados en las secciones anteriores. Así, el algoritmo de adaptación podría calcular los parámetros del control PID para cada punto de funcionamiento.

La principal dificultad en la implantación de este tipo de control radica en la elección del algoritmo de adaptación que asegure la estabilidad y correcto funcionamiento del sistema. Normalmente, este algoritmo utiliza técnicas como la del gradiente o del descenso más profundo [GASTINEL-75] para minimizar el error $\mathbf{q}_e$ entre el comportamiento del sistema real y del modelo de referencia.

De nuevo, el análisis teórico de la citada estabilidad es complejo, por lo que en la práctica es frecuente conformarse con un comprobación experimental del comportamiento del sistema. Estudios más detallados sobre el control adaptativo con un modelo de referencia y su aplicación para robots, pueden encontrarse en [ASTRÖM-89], [CRAIG-88] y [KOIVO-83].

7.3.3. Control de par calculado adaptativo

En el epígrafe 7.2 se examinó la técnica de desacoplamiento por inversión del modelo, en la cual el conocimiento de las matrices **D** y **N** del modelo dinámico del robot permitían aplicar una ley de control de la forma [7.39]:

$$\mathbf{u} = \mathbf{K}^{-1}\mathbf{D}(q)\left[s^2\mathbf{I}\mathbf{q}_d + \mathbf{R}(s)\,\mathbf{e}\right] + \mathbf{N}(q,\dot{q}) \qquad [7.40]$$

consiguiéndose que la evolución de las articulaciones respondiera a [7.36]:

$$\mathbf{q}_d = \mathbf{q} + \left(s^2\mathbf{I} + \mathbf{R}\right)^{-1}\mathbf{D}^{-1}(q)\tau_e \qquad [7.41]$$

Esto es, en ausencia de perturbaciones, coincidían la trayectoria deseada $\mathbf{q}_d$ y la real $\mathbf{q}$. Y en presencia de perturbaciones de tipo escalón, el error ($\mathbf{e} = \mathbf{q}_d - \mathbf{q}$) tendería a cero [7.37].

Los problemas de este método surgen por una parte de la dificultad de calcular en tiempo real la expresión [7.40] del par y por otra parte, del difícil conocimiento preciso del modelo dinámico del robot (matrices **D** y **N**), que además de variar con la localización **q** del robot, depende de ciertos parámetros, como masas, inercias o rozamientos, de difícil obtención.

El par calculado adaptativo trata de resolver este problema con una identificación paramétrica en línea de las matrices **D** y **N**. De este modo, el esquema de control presentado en la Figura 7.10 se convertirá en el de la Figura 7.15. De nuevo, las dificultades aparecen a la hora de asegurar la estabilidad del sistema y de su implantación en tiempo real. En concreto, la convergencia del algoritmo de identificación no está asegurada, por cuanto que los valores que toman las señales $\mathbf{q}_d$ y **q** no están seleccionados con vistas a la identificación del modelo, sino que son consecuencia de las tareas encomendadas al robot.

Asimismo, el cálculo en tiempo real de las matrices **D** y **N** es extremadamente laborioso. Un estudio detallado de la técnica de control por par calculado adaptativo y sus variantes puede verse en [LEWIS-93].

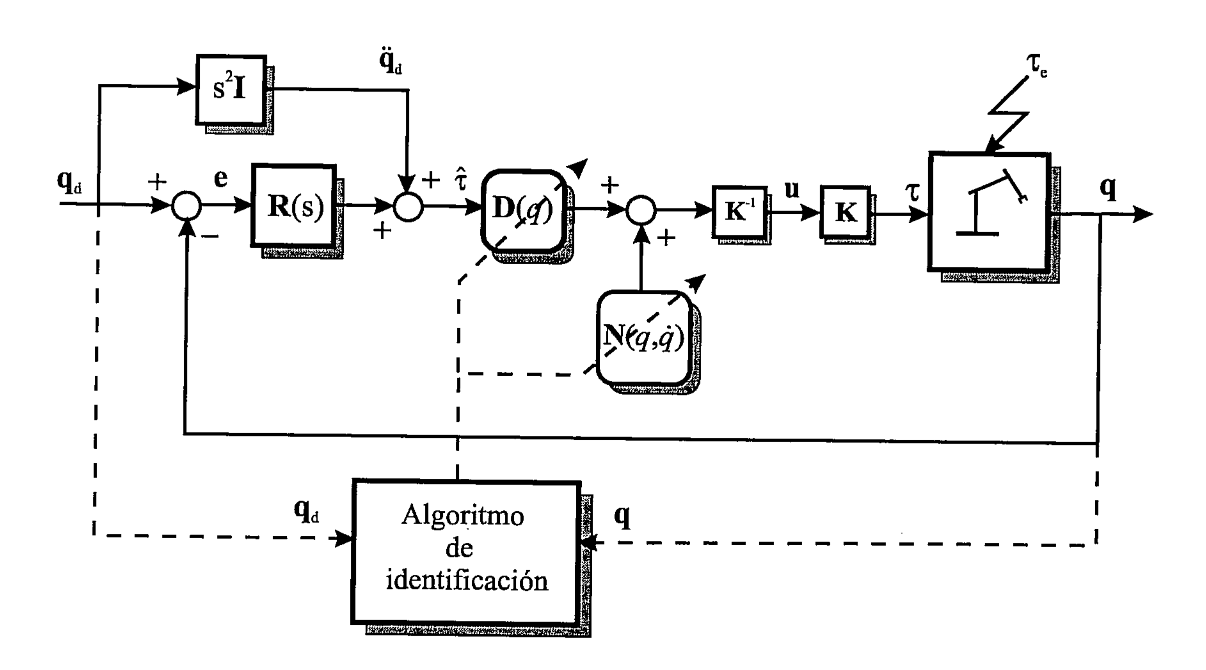


Figura 7.15. Esquema de control por par calculado adaptativo.

7.4. ASPECTOS PRÁCTICOS DE LA IMPLANTACIÓN DEL REGULADOR

Los algoritmos de control anteriormente expuestos, superadas sus limitaciones teóricas como robustez y convergencia, parecen prometedores por cuanto aseguran el objetivo de que la trayectoria deseada del robot se aproxime en mayor o menor grado a la real. Sin embargo, la implantación práctica de estos algoritmos se encuentra con diferentes tipos de problemas, que en algunas ocasiones dificultan y complican su puesta en marcha o incluso anulan completamente su utilidad.

La primera dificultad, ya citada, se deriva del elevado coste computacional implicado en el cálculo de alguno de ellos. Así, en general, las técnicas de inversión del modelo (con métodos adaptativos o no) precisan de la obtención en tiempo real de las matrices **D** y **N**. En algunos casos, la reciente potencia de los equipos de cálculo utilizando procesadores digitales de señal (DSP) o técnicas de procesamiento en paralelo (transputer), resuelven este problema.

Se van a tratar aquí otro tipo de dificultades, comunes en general a cualquier sistema de control discreto y que se centran en la implementación de las técnicas más básicas, como es la del control PID.

7.4.1. Ajuste de los parámetros del PID. Especificaciones de diseño

En cualquiera de las técnicas de control establecidas a lo largo de los epígrafes anteriores, se planteaba la existencia de un control PID cuyo objetivo era que, aún con la presencia de posibles perturbaciones, el error entre la trayectoria deseada y la real convergiese a cero de la manera más rápida.

Este regulador tenía la forma estándar dada por la expresión:

$$R(s) = k_p + \frac{k_i}{s} + k_d s \qquad [7.42]$$

quedando por decidir los valores concretos de las constantes proporcional k_p, integral k_i y derivativa k_d.

La selección de estos parámetros afectará al modo en que el robot alcanza su posición final al variar la referencia de ésta o al recibir una perturbación que trata de modificarla [PAUL-81]. En el caso simple del control monoarticular, el modelo del robot responde al esquema de la Figura 7.5.

Suponiendo que se utilicen las tres acciones del regulador PID (i.e. k_p, k_i, k_d distintas a cero) la función de transferencia del regulador podrá suponerse de la forma:

$$R(s) = k_r \frac{(s+a)(s+b)}{s} \qquad [7.43]$$

La Figura 7.16 representa el lugar de las raíces del sistema global. La elección adecuada de los ceros a y b y de la ganancia k_r (lo que equivale a seleccionar las constantes k_p, k_i, k_d) se hará de acuerdo

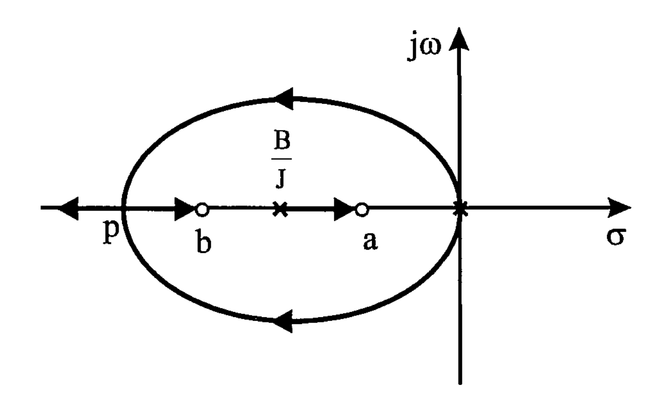


Figura 7.16. Lugar de las raíces de una articulación con control PID.

a las especificaciones de funcionamiento. En los sistemas de control se busca habitualmente una respuesta lo más rápida posible, admitiendo cierto grado de sobreoscilación.

Con estos criterios se selecciona una posición para los polos dominantes del sistema y se ajustan los valores de la ganancia k_r y del cero b para tal fin [PUENTE-79], [OGATA-93] y [BARRIENTOS-96]. La posición del cero a se selecciona de modo que compense el efecto desestabilizador originado por el polo en el origen introducido por la acción integral. En el caso de robots, si bien se persigue la máxima velocidad de respuesta, no es admisible la sobreoscilación por cuanto ésta podría originar la colisión del robot con su entorno (considérese, por ejemplo, el caso de que el robot se acerque a un objeto situado sobre una mesa). Por lo tanto, la posición de los polos dominantes debe escogerse buscando que el sistema sea críticamente amortiguado (punto p del lugar de las raíces)

Suponiendo inicialmente que sólo se utilizan acciones PD, se tendrá que el polinomio característico del sistema será (mediante deducción de la expresión [7.22]):

$$p(s) = s^2 + \frac{B + k_d}{J} s + \frac{k_p}{J} \qquad [7.44]$$

correspondiente a un sistema de 2º orden con coeficiente de amortiguamiento ξ y frecuencia natural ω_n dados por:

$$\xi = \frac{B + k_d}{2\sqrt{k_p J}} \qquad \omega_n = \sqrt{\frac{k_p}{J}} \qquad [7.45]$$

Para que el sistema sea críticamente amortiguado, el coeficiente de amortiguamiento ξ deberá ser la unidad y por tanto:

$$\frac{B+k_d}{2\sqrt{k_p J}}=1 \qquad [7.46]$$

Por otra parte, si bien se ha considerado que la estructura del robot es rígida, en realidad ésta tendrá una determinada flexibilidad estructural caracterizada por una frecuencia de resonancia estructural ω_r. En [PAUL-81] se aconseja evitar que el sistema se vea excitado por una frecuencia cercana a ω_r, por lo que la frecuencia natural del sistema regulado deberá ser inferior a la mitad de ω_r. Esto es:

$$\omega_n < \frac{\omega_r}{2} \quad \Rightarrow \quad 2\sqrt{\frac{k_p}{J}} < \omega_r \qquad [7.47]$$

Las expresiones [7.46] y [7.47] establecen 2 condiciones que, supuestos conocidos J, B y ω_r, permiten seleccionar en primera aproximación las constantes proporcional k_p y derivativa k_d del regulador. Un ajuste más preciso de las mismas deberá realizarse por ensayo y error. En cuanto a la constante integral, como es bien sabido, tiene por objetivo anular el error permanente ante cambios bruscos (en escalón) en la perturbación τ_p o en q_d. Si bien cualquier valor de k_i distinto de cero conseguirá este objetivo (ver expresión [7.17]) una inadecuada selección de la misma, cambiaría radicalmente los logros obtenidos con el ajuste de k_p y k_d (referidos a ξ y ω_n), por cuanto la acción integral introduce una pareja polo-cero en el sistema, que dejará de comportarse como un sistema de 2º orden puro.

7.4.2. Saturación de los amplificadores. Efecto de la acción integral

Inevitablemente, los amplificadores o dispositivos electrónicos destinados a suministrar la potencia a los actuadores (eléctricos o hidráulicos) del robot, tienen un margen de actuación limitado. Así, aun en el caso de comportamiento ideal en el que la salida de la etapa de potencia o amplificador es proporcional a la señal de consigna, esta proporcionalidad es válida únicamente entre unos valores extremos mínimo y máximo, superados los cuales la salida del amplificador se mantiene constante e independiente de la

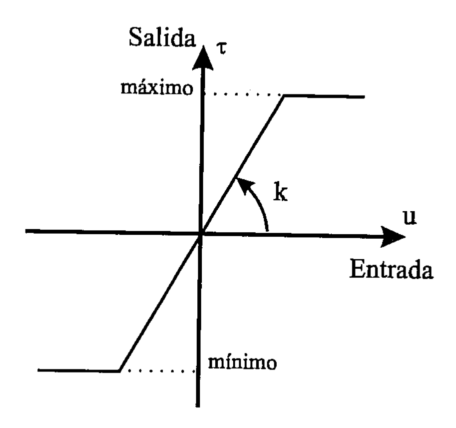


Figura 7.17. Relación entre entrada y salida de un amplificador con saturaciones.

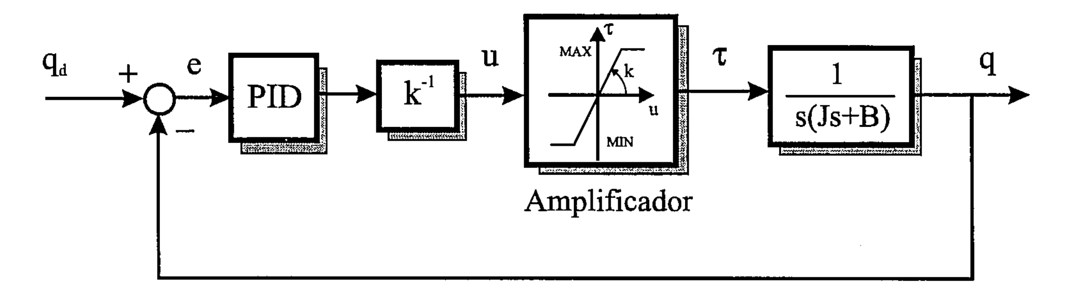


Figura 7.18. Control PID monoarticular con saturación en el amplificador.

entrada[1] (Figura 7.17). Como se ha visto en los epígrafes anteriores, los esquemas de control utilizados para realizar el control de las articulaciones del robot, utilizan de manera general el error entre trayectoria deseada y real para obtener la señal de mando, que será entrada de la etapa de potencia que alimenta a los actuadores.

La presencia de errores mantenidos, junto a la utilización de la acción integral, puede originar que la señal de mando para los amplificadores supere los límites del comportamiento lineal, siendo entonces la señal de mando a los actuadores constante, independiente del error que se tenga en ese momento. Puede decirse entonces que el sistema no utiliza la realimentación, pues la señal de mando permanece constante independientemente de la salida.

Más aún, aunque la posición real q igualase o incluso superase a la deseada q_d y, por lo tanto, el error fuese nulo o negativo, la acción integral del regulador haría que su salida se mantuviese durante cierto tiempo en un valor positivo suficientemente elevado como para que la señal de mando del actuador fuese la máxima correspondiente a la saturación. Sólo cuando el error se mantuviese negativo durante un tiempo suficientemente largo, el amplificador entraría en su zona lineal. Este fenómeno, denominado en la literatura en lengua inglesa *wind-up*, origina mayores oscilaciones en la posición real q, tardando más en estabilizarse.

La Figura 7.18 representa el caso simple de control PID monoarticular en el que el amplificador de ganancia k presenta una saturación caracterizada por los valores máximo y mínimo de su salida (MAX y MIN, respectivamente). En la Figura 7.19 y Figura 7.20 se representa la salida q y el par proporcionado τ, respectivamente, para los casos de sistema sin saturación y sistema con unos límites de saturación dados por MAX = +1 y MIN = -1. En ambos casos se ha tomado k = 1, J = 1 y B = 20. El regulador PID se ha fijado a:

$$R(s) = 1 + \frac{1}{s}$$

[1] En la práctica este comportamiento no es lineal ni tan siquiera entre los valores máximos y mínimos, existiendo zonas muertas, histéresis y otras no linealidades.

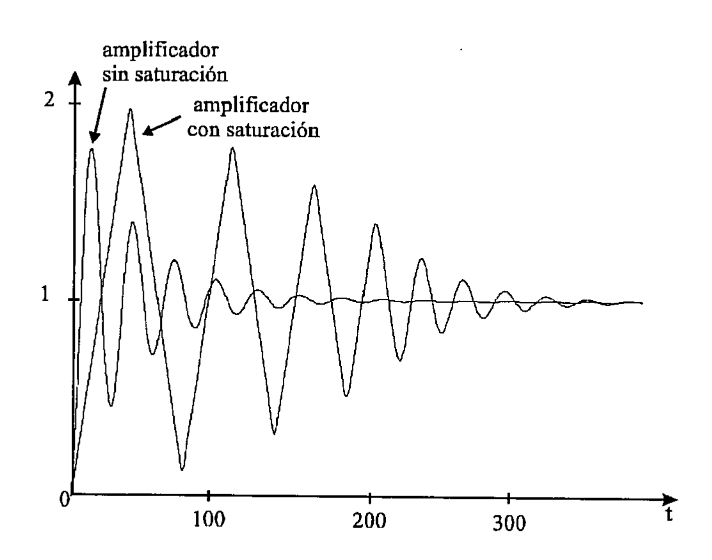


Figura 7.19. Comparación de respuesta con y sin saturación en amplificador.

sin incluir la acción derivativa, que agrava notoriamente el efecto de la saturación por cuanto provoca valores de señal elevados a la salida del regulador. Se observa en ellas cómo la saturación origina mayores oscilaciones en q y un mayor tiempo en alcanzar el valor final. También se observa como el par de mando τ permanece dentro de su comportamiento no lineal durante buena parte del tiempo.

Existen en la práctica diversas técnicas para evitar las consecuencias derivadas del funcionamiento de la saturación del amplifiacador [ASTRÖM-95]. Posiblemente, el más utilizado en sistemas de regula-

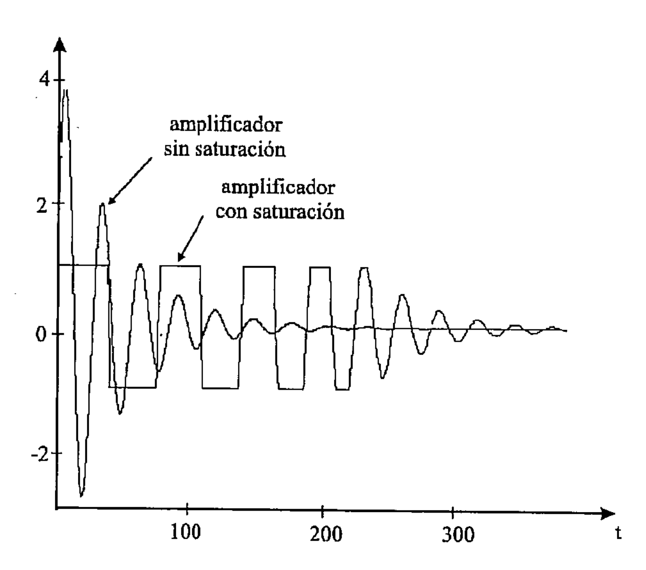


Figura 7.20. Comparación entre el par τ proporcionado por un amplificador con y sin saturación.

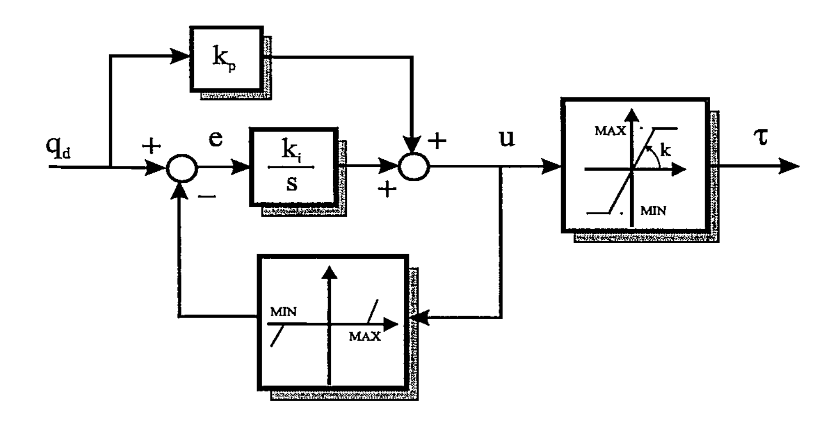


Figura 7.21. Control monoarticular con regulador PID *antiwind-up*.

ción de posición y en particular en robótica, es el de anular la acción integral hasta que el error entre q y q_d no entre dentro de los límites establecidos, esto es, se esté llegando a la posición de destino.

Otra alternativa es la mostrada en la Figura 7.21. El sistema mostrado, conocido como regulador PI con *antiwind-up*, corta la acción integral en cuanto se satura la salida del integrador [FRANKLIN-91].

También una modificación en el esquema de control PID permite reducir las ocasiones en las que el amplificador se satura. Así, cuando se produce un cambio brusco entre la posición deseada y la real (cambio de consigna, por ejemplo), la acción proporcional y muy especialmente la derivativa, originan a la salida del regulador fuertes acciones de control que pueden originar la saturación del amplificador. Para evitar esta circunstancia puede recurrirse al esquema de control de la Figura 7.22 denominado I-PD [OGATA- 93]. En este esquema, las acciones proporcional y derivativa se han pasado a la realimentación, siendo la función de transferencia (mientras se trabaja en la zona lineal) entre la posición real y la deseada:

$$\frac{q(s)}{q_d(s)} = \frac{\frac{k_i}{s}}{(Js+B)s+\left(k_p+\frac{k_i}{s}+k_d s\right)} \qquad [7.48]$$

que como se observa difiere en el numerador de la correspondiente función de transferencia entre q y q_d para el caso de regulador PID (ecuación [7.21]), teniendo el mismo polinomio característico.

En general el control I-PD disminuye el problema de la saturación, pero da lugar a respuestas más lentas.

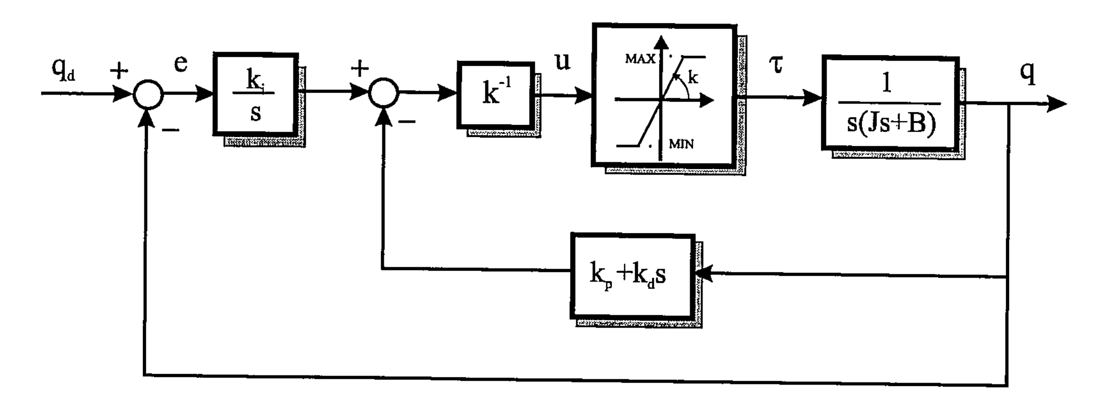


Figura 7.22. Control I-PD monoarticular.

7.4.3. Flexión y oscilaciones estructurales

Como se ha indicado en el epígrafe 7.4.1, la flexibilidad estructural de los eslabones y articulaciones del robot pueden originar oscilaciones no deseables. Estas oscilaciones no pueden ser en principio tratadas por el sistema de control, por cuanto, en general, no son medibles al estar los sensores de posición típicamente situados en las articulaciones y no a lo largo de los eslabones. En el caso de robots de grandes dimensiones o de estructura ligera, este problema pasa a un primer plano, siendo necesario considerar la flexión estructural tanto a nivel de control cinemático como dinámico [AMARIS-95].

Existen diversas técnicas destinadas a disminuir los efectos de la flexión y las oscilaciones estructurales. Algunos de ellos precisan de la utilización de sensores situados fuera de la articulación, ya sea a lo largo del eslabón o en el extremo del robot (a estos sensores se les denomina *no colocados*). La medida proporcionada por estos sensores se utiliza para generar acciones de control que compensen las oscilaciones de la estructura. De manera alternativa o complementaria, se puede tratar de evitar, o cuanto menos disminuir, el problema en su origen, evitando excitar a los actuadores del robot con señales que originen oscilaciones y compensando *a priori* las flexiones estáticas previstas.

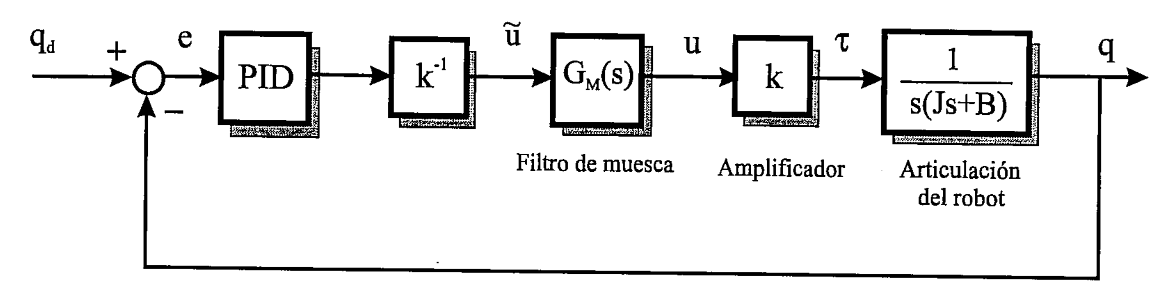


Figura 7.23. Control monoarticular con la inclusión de un filtro de muesca.

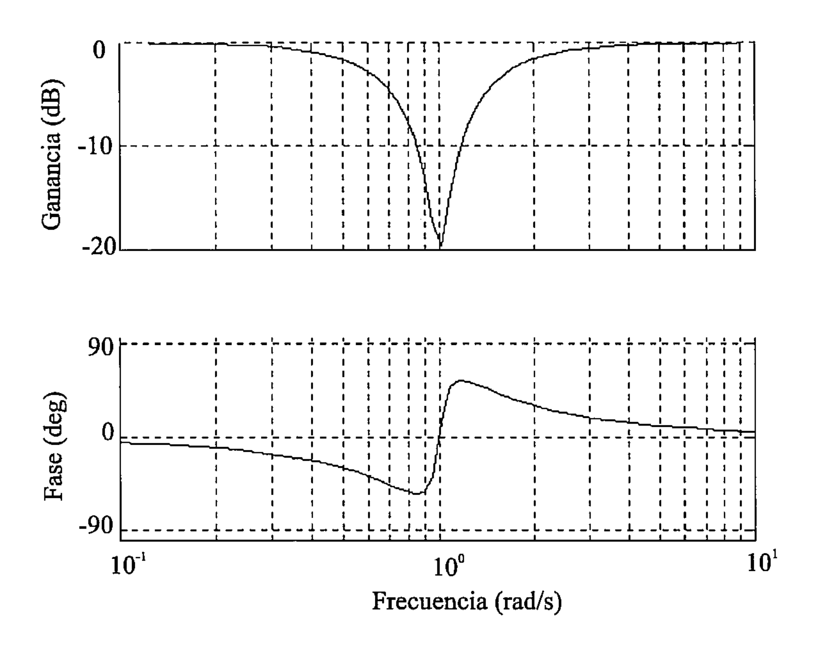


Figura 7.24. Diagrama de Bode de un filtro de muesca.

Con tal propósito, ya se han indicado ciertas recomendaciones para la selección de los parámetros del PID. Otra técnica frecuente es la utilización de un filtro a la salida del regulador que elimine de la señal de mando las frecuencias próximas a la de resonancia mecánica del sistema. Este filtro, denominado filtro de muesca (o *notch filter*) [ROHRS-94] origina una fuerte atenuación en un estrecho rango de frecuencias. Este efecto se consigue mediante el esquema de control de la Figura 7.23 (caso monoarticular), en el que se ha incorporado el filtro de muesca $G_M(s)$ entre la acción de control $\tilde{u}$ y la señal de mando que finalmente llega al sistema de accionamiento.

Este filtro tiene típicamente por función de transferencia:

$$G_M(s) = \frac{s^2 + 2\xi\omega_n s + \omega_n^2}{s^2 + \omega_n s + \omega_n^2} \qquad [7.49]$$

Es decir, se compone de dos ceros y de dos polos, ambos complejos conjugados. La selección de ω_n y ξ se hace de tal modo que ω_n coincida con la frecuencia que se desea eliminar (frecuencia de resonancia estructural ω_r de la articulación considerada). En cuanto a ξ, determina la atenuación introducida a esta frecuencia, que toma el valor -20 log(2ξ).

La Figura 7.24 representa los diagramas de Bode de un filtro de muesca, pudiéndose observar el efecto del filtro selectivo conseguido (se ha tomado $\omega_n = 1$ y $\xi = 0.05$).

7.5. BIBLIOGRAFÍA

[AMARÍS-95] H. Amarís, *Modelado y control de la flexión de robots de grandes dimensiones*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1995.

[ASTRÖM-95] K.J. Aström y Hägglund, *PID Controllers: Theory, Design and Tuning*, ISA, 1995.

[ASTRÖM-89] K.J. Aström, K.B. Wittenmark, *Adaptive Control*, Addison-Wesley, 1989.

[BARRIENTOS-96] A. Barrientos et al., *Control de sistemas continuos. Problemas resueltos*, McGraw-Hill, Madrid, 1996.

[CRAIG-88] J.J. Craig, *Adaptive Control of Mechanical Manipulators*, Addison-Wesley, 1988.

[FRANKLIN-91] G.F. Franklin, J.D. Powell y A. Emami-Naeini, "Principios esenciales de la retroalimentación", *Control de sistemas dinámicos con retroalimentación,* Addison-Wesley, 1991.

[GASTINEL-75] N. Gastinel, *Análisis numérico lineal*, Editorial Reverté, 1975.

[KOIVO-89] A.J. Koivo, *Fundamentals for control of robot manipulators*, John Wiley & Sons, 1989.

[KOIVO-83] A. J. Koivo, "Adaptive linear controller for robot manipulators", *IEEE Transactions on Automatic Control*, AC-28, pp. 162-171, 1983.

[LEWIS-93] F.L. Lewis, C.T. Abdallah y D.M. Dawson, *Control of Robot Manipulators*, Macmillan Publishing Co., Nueva York, 1993.

[OGATA-93] K. Ogata, *Ingeniería de Control Moderna*, Prentice-Hall Hispanoamericana, 2ª edición,1993.

[PAUL-81] R. P Paul, "Static Forces", en: *Robot Manipulators, Mathematics, Programming and Control*, The MIT Press, Massachussets, 1981.

[PUENTE-79] E.A. Puente, *Regulación automática I*, Sección Publicaciones, ETSI Industriales, Universidad Politécnica de Madrid, Madrid 1979.

[RODRIGUEZ-96] F. Rodríguez y M.J. López, *Control adaptativo y robusto*, Secretaría de Publicaciones de la Universidad de Sevilla, Sevilla, 1996.

[ROHRS-94] Ch. E. Rohrs, J.L. Melsa y D.G. Schultz, "Diseño de sistemas de control", en: *Sistemas de control lineal*, Mc-Graw Hill, 1994.

Capítulo 8

PROGRAMACIÓN DE ROBOTS

Quien hace un cesto, hará ciento.
Refranero popular

Un robot industrial es básicamente un manipulador multifuncional reprogramable, siendo esta capacidad de reprogramación la que permite su adaptación de una manera rápida y económica a diferentes aplicaciones. La programación de un robot se puede definir como el proceso mediante el cual se le indica a éste la secuencia de acciones que deberá llevar a cabo durante la realización de su tarea. Estas acciones consisten en su mayor parte en moverse a puntos predefinidos y manipular objetos del entorno.

Durante la ejecución de un programa se interacciona con la memoria del sistema, leyendo y actualizando el contenido de las variables utilizadas en el programa; con el sistema de control cinemático y dinámico del robot, encargados de dar la señal de mando a los accionamientos del robot a partir de las especificaciones del movimiento que se les proporciona; y con las entradas-salidas del sistema, consiguiéndose así la sincronización del robot con el resto de las máquinas y elementos que componen su entorno.

El sistema de programación es, por tanto, la herramienta con que cuenta el usuario para acceder a las diversas prestaciones del robot, existiendo una relación directa entre las características y posibilidades del sistema de programación y las del robot en sí mismo.

En los epígrafes siguientes se comienza examinando los diferentes procedimientos existentes para la programación de robots, realizándose seguidamente una clasificación de los mismos. A continuación se analizan aquellas características propias de un sistema de programación para robots, y que les distingue de otro tipo de sistemas de programación, como son los de equipos informáticos. Estos aspectos se concretan posteriormente en un caso práctico de programación de un robot industrial. Finalmente, se describen brevemente dos sistemas actuales de programación de robots industriales: RAPID de ABB y V+ de Adept Technology.

8.1. MÉTODOS DE PROGRAMACIÓN DE ROBOTS. CLASIFICACIÓN

Programar un robot consiste en indicar paso por paso las diferentes acciones (moverse a un punto, abrir o cerrar la pinza, etc.) que éste deberá realizar durante su funcionamiento automático. La flexibilidad en la aplicación del robot y, por tanto, su utilidad dependerá en gran medida de las características de su sistema de programación.

En la actualidad no existe normalización alguna en cuanto a los procedimientos de programación de robots. Por el contrario, cada fabricante ha desarrollado su particular método, válido únicamente para sus propios robots. Sin embargo, existen algunos sistemas de programación que han servido de modelo para el desarrollo de otros. Tal es el caso del lenguaje AL [FINKEL-74] desarrollado por la Universidad de Stanford en los años setenta y que ha servido de referencia para muchos de los sistemas comerciales existentes. También es necesario indicar que, a pesar de la variedad de sistemas de programación existentes en el mercado, en todos ellos se dan una serie de características comunes y que se explicarán con detalle en el epígrafe 8.2.

Los escasos intentos de unificar en cierta medida los procedimientos de programación de robots no han tenido hasta la fecha el reconocimiento y la aceptación necesarios. De entre estos, tal vez el más destacable haya sido el denominado IRDATA [REMBOLD-85] (reconocido como estándar por la VDI alemana), concebido por la Universidad de Karlsruhe como un código intermedio entre el sistema de programación utilizado y el propio sistema de programación del robot.

Existen diversos criterios para realizar una clasificación de los métodos de programación de robots. Algunas atienden a la potencia del método, mientras que otras clasificaciones hacen referencia al sistema empleado para indicar la secuencia de acciones a realizar. Este segundo criterio es más ilustrativo a la hora de dar a conocer las alternativas existentes para programar un robot, siendo por esta razón el que se va a seguir.

Según este criterio, un robot puede ser programado mediante lo que se denomina **guiado** o mediante un procedimieno **textual**, existiendo robots que conjugan ambos tipos. Este primer nivel de división admite un segundo, atendiendo a la potencia y particularidades del método de programación. Las caracteríticas de cada procedimiento de programación serán más o menos ventajosas según la aplicación a la que se destina el robot.

8.1.1. Programación por guiado

La programación por guiado o aprendizaje consiste en hacer realizar al robot, o a una maqueta del mismo, la tarea (llevándolo manualmente por ejemplo) al tiempo que se registran las configuraciones adoptadas, para su posterior repetición de manera automática.

Para guiar al robot por las trayectorias o puntos deseados se utilizan diferentes soluciones. Si los actuadores del robot están desconectados y es el programador quien debe aportar directamente la energía para mover el robot, se trata de un guiado pasivo. En este caso, el programador puede tomar el extremo del robot y llevarlo hasta los puntos deseados siguiendo las trayectorias más adecuadas. La unidad de

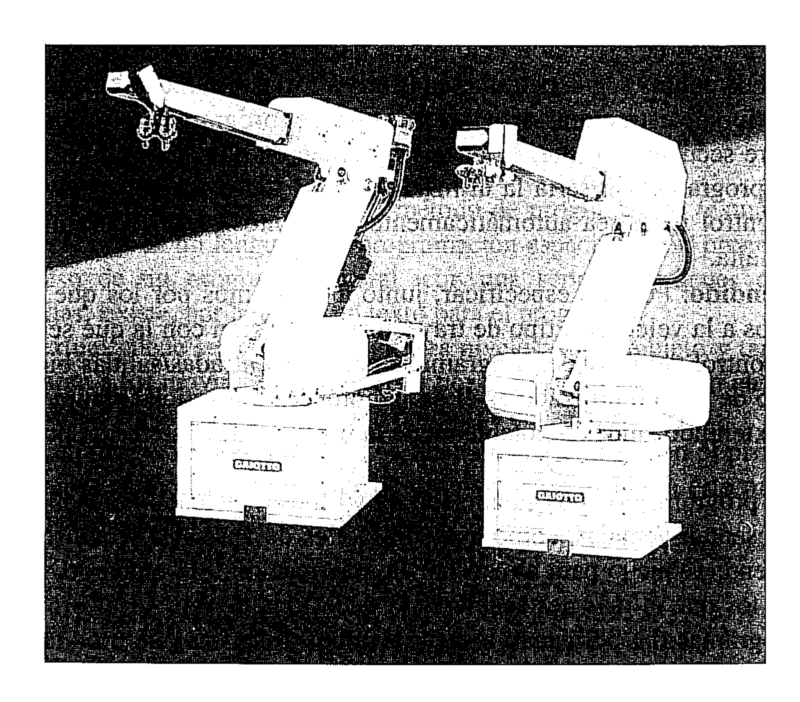


Figura 8.1. Robots de pintura RI/93 (hidráulico) y RE/91 (eléctrico) de programación mediante guiado pasivo directo. (Cortesía de Gaiotto.)

control del robot registrará, de manera automática, la señal de los sensores de posición de las articulaciones en todos los puntos recorridos. Este tipo de guiado se denomina guiado pasivo directo. Un ejemplo de robot industrial programado por este procedimiento se encuentra en los robots de pintura de la firma Gaiotto (Figura 8.1).

La dificultad física de mover toda la estructura del robot se resuelve con la utilización del guiado pasivo por maniquí. En este caso se dispone de un doble del robot mientras que éste permanece fuera de línea. El maniquí posee idéntica configuración que el robot real, pero es mucho más ligero y fácil de mover. La programación se efectúa llevando de la mano a este doble, mientras que la unidad de control muestrea y almacena, con determinada frecuencia, los valores que toman los sensores de posición de las articulaciones, para su posterior repetición por el robot. Los robots, también de pintura, fabricados por Nordson se programan utilizando este procedimiento.

Frente a estos sistemas pasivos, otra posibilidad permite emplear el propio sistema de accionamiento del robot, controlado desde una botonera o un bastón de mando (joystick) para que sea éste el que mueva sus articulaciones. Se dirá entonces que se trata de un guiado activo. Ejemplos de este tipo se encuentran en los robots de la antes ASEA y ahora ABB (ARLA) o en los de Cincinnati Milacron (T3). En este caso, lo habitual es que la unidad de control únicamente registre aquellas configuraciones del robot que el programador indique expresamente.

Atendiendo a la potencia del sistema, se puede hablar de guiado básico y de guiado extendido.

- **Guiado básico**: El robot es guiado consecutivamente por los puntos por los que se quiere que pase durante la fase de ejecución automática del programa. Durante ésta, la unidad de control interpola dichos puntos según determinadas trayectorias. En muchas ocasiones no es posible incluir ningún tipo de estructuras de control dentro del programa, por lo que los puntos son recorridos siempre secuencialmente, en el mismo orden en que se programaron. Un caso límite de este tipo de programación sería la utilizada en casi todos los robots de pintura, en los que la unidad de control muestrea automáticamente los puntos recorridos por el robot con una frecuencia muy alta.
- **Guiado extendido**: Permite especificar, junto a los puntos por los que deberá pasar el robot, datos relativos a la velocidad, tipo de trayectoria, precisión con la que se desea que se alcancen los puntos, control del flujo del programa, atención a entradas/salidas binarias, etc. En este caso, el método de guiado utilizado es el de la botonera o joystick (Figura 8.2). La programación por guiado extendido aumenta notablemente la potencia del sistema de programación.

Los métodos de programación por guiado son muy útiles e incluso imprescindibles en muchas ocasiones. Además presentan ventajas, como que son fáciles de aprender y que requieren un espacio de memoria relativamente pequeño para almacenar la información. Sin embargo, presentan una serie de inconvenientes, de los que el más destacable es la necesidad de utilizar al propio robot y su entorno para realizar la programación, obligando a sacar al robot de la línea de producción e interrumpiendo ésta. Otros inconvenientes frecuentes son la inexistencia de una documentación del programa y la

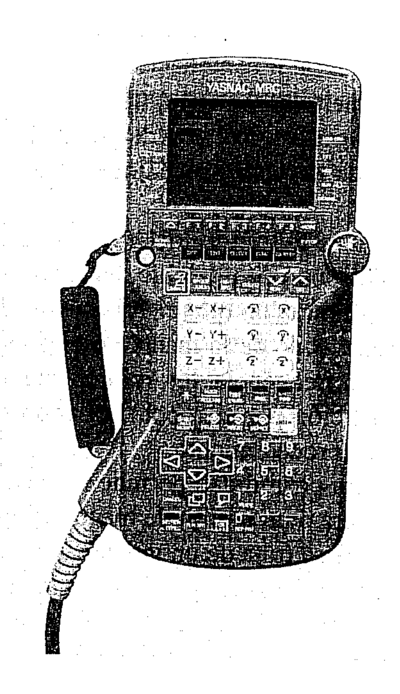


Figura 8.2. Botonera o paleta de programación de un robot industrial. (Cortesía de Yaskawa.)

dificultad de realizar modificaciones en el mismo, inconvenientes ambos que conducen a una difícil depuración y puesta a punto de las aplicaciones.

8.1.2. Programación textual

Como alternativa a la programación por guiado, el método de programación textual permite indicar la tarea al robot mediante el uso de un lenguaje de programación específico. Un programa se corresponde ahora, como en el caso de un programa general, con una serie de órdenes que son editadas y posteriormente ejecutadas. Existe, por lo tanto, un texto para el programa.

La programación textual puede ser clasificada en tres niveles: robot, objeto y tarea, dependiendo de que las órdenes se refieran a los movimientos a realizar por el robot, al estado en que deben ir quedando los objetos manipulados o al objetivo (o subobjetivo parcial) a conseguir.

En el estado actual, la programación de robots se queda materialmente en el primero de ellos (nivel robot), existiendo una gran cantidad de lenguajes de programación textual a este nivel [BONNER-82], de entre los que se pueden destacar por orden cronológico:

- AL (Universidad de Stanford - 1974) [FINKEL-74].
- AML (IBM - 1979) [TAYLOR-82].
- LM (Universidad de Grenoble, SCEMI - 1981).
- VAL II (UNIMATION - 1983) [STAÜBLI-92].
- V+ (ADEPT - 1989) [ADEPT-89].
- RAPID (ABB -1994) [ABB-94].

Se han realizado diversos intentos de desarrollar lenguajes a nivel objeto [KOUTSOU-81], pero las dificultades con que se han encontrado los investigadores han impedido una implementación eficiente del lenguaje. Como ejemplos pueden citarse:

- LAMA (MIT - 1976) [LOZANO-76].
- AUTOPASS (IBM - 1977) [LIEBERMAN-77].
- RAPT (Universidad de Edimburgo - 1978) [POPPLESTONE-78] [AMBLER-86].

Para aclarar el modo en que se programaría un robot en cada uno de los niveles anteriormente mencionados, se va a resolver una tarea concreta utilizando las tres alternativas.

La Figura 8.3 representa el robot y su entorno. Se pretende situar la pieza A, sobre la que se apoya la pieza B, en el interior del orificio de la pieza D. A continuación se presenta el programa en los tres niveles de manera simplificada y utilizando lenguajes hipotéticos.

- **Nivel robot**: Es necesario especificar cada uno de los movimientos que ha de realizar el robot, así como la velocidad, direcciones de aproximación y salida, apertura y cierre de la pinza, etc. Será necesario también descomponer la tarea global en varias subtareas, como por ejemplo, *quitar B de A (poniendo B sobre C)*, *introducir A en D*, etc. Para el ejemplo que se está considerando, la operación que consiste en colocar B sobre C tendría el siguiente aspecto:

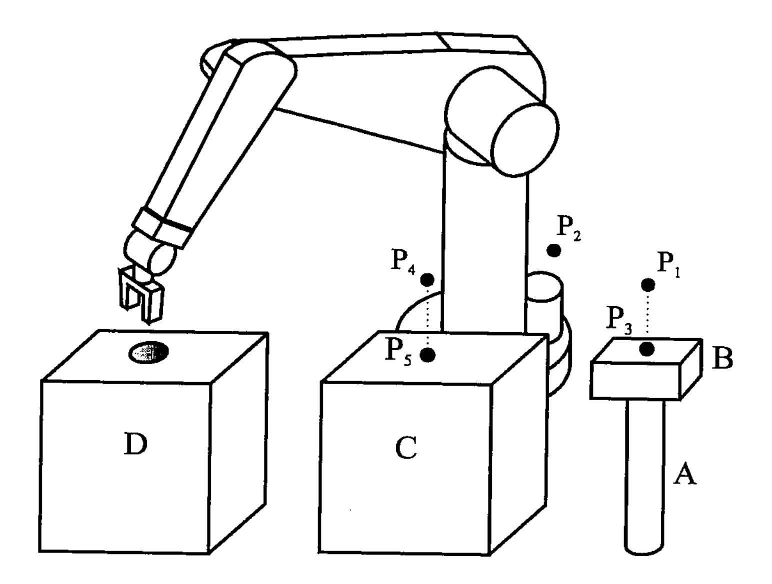


Figura 8.3. Ejemplo de tarea de programación.

```
Mover_a P1 via P2            ; Situarse en un punto sobre la pieza B
Vel = 0.2 * VELMAX           ; Reducir la velocidad
Pinza = ABRIR                ; Abrir la pinza
Prec = ALTA                  ; Aumentar la precisión
Mover_recta_a P3             ; Descender verticalmente en línea recta
Pinza = CERRAR               ; Cerrar la pinza para coger la pieza B
Espera= 0.5                  ; Esperar para garantizar cierre de pinza
Mover_recta_a P1             ; Ascender verticalmente en línea recta
Prec = MEDIA                 ; Decrementar la precisión
Vel = VELMAX                 ; Aumentar la velocidad
Mover_a P4 via P2            ; Situarse sobre la pieza C
Prec = ALTA                  ; Aumentar la precisión
Vel = 0.2 * VELMAX           ; Reducir velocidad
Mover_recta_a P5             ; Descender verticalmente en línea recta
Pinza = ABRIR                ; Abrir pinza
```

- **Nivel Objeto**: Disminuye la complejidad del programa. La programación se realiza de manera más cómoda, ya que las instrucciones se dan en función de los objetos a manejar. Un planificador de la tarea se encargará de consultar una base de datos y generar las instrucciones a nivel de robot.

```
Situar B sobre C haciendo coindidir
LADO_B1 con LADO_C1 y LADO_B2 con LADO_C2 ;
Situar A dentro D haciendo coincidir
EJE_A con EJE_HUECO_D y BASE_A con BASE_D ;
```

- **Nivel Tarea** : El programa se reduce a una única sentencia ya que se especifica qué es lo que debe hacer el robot en lugar de cómo debe hacerlo.

```
Ensamblar A con D
```

Es importante destacar que, cada vez con mayor notoriedad, los sistemas de programación de robots tienden a combinar los dos modos básicos (guiado y textual), permitiéndose desarrollar el programa mediante la escritura de las instrucciones, y utilizando la posibilidad de guiado en línea en aquellos momentos en que sea necesario. Sistemas como RAPID de ABB, VAL II de Staübli y V+ de Adept Technology son ejemplos de esta ambivalencia.

Como resumen, la Figura 8.4 recoge los diferentes métodos de programación de robots existentes.

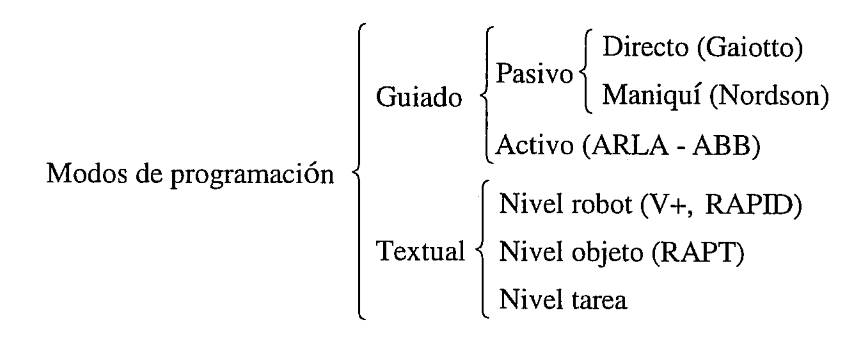


Figura 8.4. Métodos de programación de robots.

8.2. REQUERIMIENTOS DE UN SISTEMA DE PROGRAMACIÓN DE ROBOTS

A pesar de la falta de normalización entre los métodos de programación de robots existentes, las necesidades comunes han originado un cierto paralelismo y afinidad entre casi todos ellos. Esta circunstancia permite establecer una serie de características generales que se manifiestan en los elementos de programación que contienen. Tradicionalmente los requerimientos generales que se vienen estableciendo [CRAIG-89] [LOZANO-82] [BLUME-86] para un sistema de programación de robots son los siguientes:

- Entorno de programación.
- Modelado del entorno.
- Tipo de datos.
- Manejo de entradas/salidas (digital y analógica).

- Control de movimiento.
- Control del flujo de ejecución del programa.

Cada sistema de programación de robots da respuesta a estos requerimientos en mayor o menor grado. Así, mientras algunos no contemplan modelo alguno del entorno, otros basan totalmente su modo de funcionamiento en la existencia y mantenimiento de un detallado modelo geométrico.

Se realiza a continuación una descripción de estos requerimientos, prestando mayor atención a los que se consideran más frecuentes.

8.2.1. Entorno de programación

Como en cualquier lenguaje de programación convencional, es de gran importancia, para conseguir un aumento considerable de productividad de la programación, el contar con un entorno de programación adecuado a las necesidades.

Programar las acciones de un manipulador es complicado en el sentido de que en todo momento existe una interacción con el entorno, tratándose de un proceso continuo de prueba y error. Esta es la principal causa que lleva a que la mayoría de los sistemas de programación de robots sean de tipo interpretado, pudiéndose realizar un seguimiento paso a paso de lo programado en cada momento. Se evita así el tedioso ciclo de editar-compilar-ejecutar muy costoso en tiempo.

Es pues importante y de desear que un sistema de programación de robots presente una buena capacidad de depuración y de ejecución paso a paso, teniendo especialmente en cuenta la interacción en tiempo real que existe entre los distintos equipos de la célula con el controlador del robot. También es aconsejable la existencia de una buena monitorización continua del desarrollo del programa [VOLZ-88].

Algunos de los sistemas de programación actuales se soportan sobre sistemas operativos multitarea, permitiendo el control simultáneo y sincronizado de varios robots o del robot con otros sistemas (visión, transportes, ect.).

8.2.2. Modelado del entorno

El modelo del entorno es la representación que tiene el robot de los objetos con los que interacciona. Normalmente este modelo se limita a características geométricas: posición y orientación de los objetos, y en ocasiones a su forma, dimensiones, peso, etc.

Para definir la posición y orientación de los objetos del modelo, lo más frecuente es asignar a cada objeto de manera solidaria un sistema de referencia, de manera que la posición y orientación de este sistema referidos a un sistema base, normalmente denominado sistema del mundo, definen de manera única las del objeto.

Algunos modelos del entorno permiten establecer relaciones entre objetos [FINKEL-74] [RODRIGUEZ-94]. Éstas establecen la posible unión física entre los objetos. Dos objetos pueden ser independientes (el movimiento de uno no afecta al otro), tener dependencia de unión rígida (el movimiento de uno implica el del otro y viceversa) o tener una dependencia de unión no rígida (el movimiento de uno implica el del otro, pero no al revés). Este modelo relacional, una vez definido es actualizado de manera automática durante la ejecución del programa mediante una estructura arborescente (Figura 8.5), simplificando notablemente la tarea del programador.

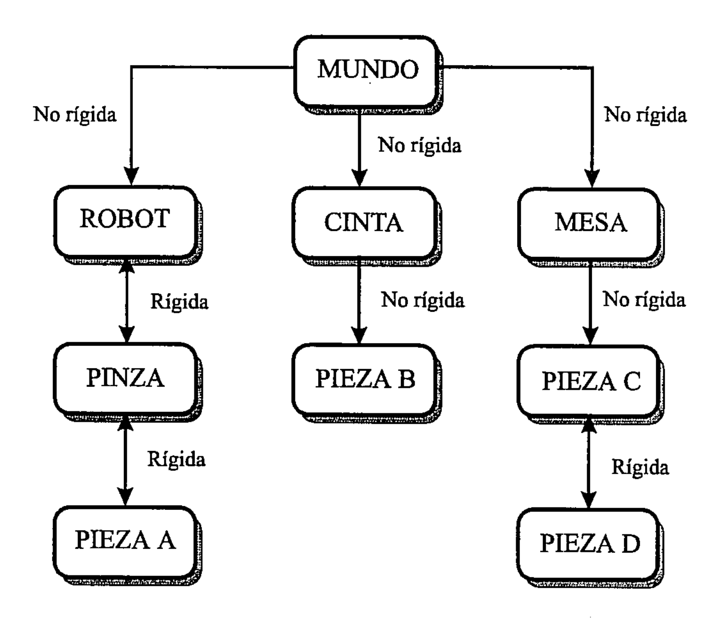


Figura 8.5. Relación entre diversos elementos de un entorno de programación.

8.2.3. Tipos de datos

Un sistema de programación de robots cuenta, además de con los tipos de datos convencionales (enteros, reales, booleanos, etc.) con otros específicamente destinados a definir las operaciones de interacción con el entorno, como son, por ejemplo, los que especifican la posición y orientación de los puntos y objetos a los que debe acceder el robot. Como ya se vio en el Capítulo 3, correspondiente a herrramientas matemáticas, la posición y orientación espacial de un objeto puede ser especificada de diversas formas.

Utilizando coordenadas articulares o del robot: mediante una n-upla $(q_1...q_n)$, donde n es el número de grados de libertad del robot, que indica los valores articulares que debe tomar el robot para posicionar y orientar su extremo como el objeto en cuestión.

Utilizando coordenadas cartesianas o del usuario: en este caso, una vez asociado un sistema de referencia de coordenadas$\{S_1\}$ al objeto, su posición se describe por las coordenadas cartesianas del origen de $\{S_1\}$, (p_x, p_y, p_z) mientras que la orientación admite diferentes tipos de representación:

- Ángulos de Euler: (α,β,γ).
- Cuaternios: $(\cos_{\theta/2}, h_x \,\mathrm{sen}_{\theta/2}, h_y \,\mathrm{sen}_{\theta/2}, h_z \,\mathrm{sen}_{\theta/2})$.
- Matriz noa:

$$\begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$$

La representación conjunta de posición y orientación del extremo del robot se consigue agrupando las tres coordenadas de posición con alguno de los métodos de representación de la orientación. Los diferentes sistemas de programación existentes emplean una o varias de estas representaciones:

- VAL II (Staübli) — emplea $(q_1 ... q_6)$, $(p_x, p_y, p_z, \alpha, \beta, \gamma)$
- AML (IBM) — emplea $(p_x, p_y, p_z, \alpha, \beta, \gamma)$
- ARLA (ABB) — emplea $(p_x, p_y, p_z, \cos_{\theta/2}, h_x \text{sen}_{\theta/2}, h_y \text{sen}_{\theta/2}, h_z \text{sen}_{\theta/2})$
- RAPID (ABB) — emplea $(p_x, p_y, p_z, \cos_{\theta/2}, h_x \text{sen}_{\theta/2}, h_y \text{sen}_{\theta/2}, h_z \text{sen}_{\theta/2})$
- V+ (Adept) — emplea $(p_x, p_y, p_z, \alpha, \beta, \gamma)$
- AL (Stanford) — emplea $(q_1 ... q_6)$ y matrices de transformación homogénea.

En VAL II, V+ y RAPID también existe la posibilidad de emplear matrices de transformación.

8.2.4. Manejo de entradas-salidas

La comunicación del robot con otras máquinas o procesos que cooperan con él, es fundamental para conseguir su integración y sincronización en los procesos de fabricación. Esta comunicación se consigue, en el nivel más sencillo, mediante señales binarias de entrada y salida. Mediante ellas el robot puede decidir comenzar una determinada acción o indicar a un dispositivo externo que comience la suya.

Para el manejo de las salidas binarias el robot posee instrucciones de activación o desactivación de las mismas. En cuanto a las entradas, el robot tiene capacidad de leerlas y controlar el flujo del programa en función de su valor, como esperas o saltos condicionados. Algunos sistemas incluyen la posibilidad de comenzar automáticamente su ciclo de trabajo cuando le llega una señal binaria determinada.

Una utilización especial de las entradas binarias es la generación de interrupciones. En estos casos una determinada señal o una combinación lógica de las mismas, se monitoriza de manera automática. Si la citada condición se verifica, la unidad de control del robot ejecuta una rutina especial, interrumpiéndose el flujo normal del programa. Esta interrupción puede realizarse inmediatamente (incluso en mitad del movimiento del robot), cuando finaliza la ejecución de la instrucción en curso (al finalizar el movimiento), o cuando finaliza el ciclo de trabajo fijado por el programa (al finalizar la secuencia).

Una utilización a mayor nivel de la comunicación del robot con su entorno lo constituye el empleo de comunicaciones mediante red local o conexión punto a punto (normalmente mediante línea serie asíncrona). Esta comunicación permite integrar al robot en un sistema informático general controlando o supervisando su funcionamiento desde un computador externo. Ejemplos de sistemas con esta posibilidad son los sistema RAPID de ABB y VAL II de Staübli.

Otra aplicación importante de las entradas-salidas del robot, ya sean digitales o analógicas, es la integración de sensores, incorporando la información de éstos al desarrollo de la tarea. Los sensores permiten ante todo realizar determinadas aplicaciones en un entorno no conocido de forma total sin tener que hacer uso de herramientas especiales. La información proporcionada por los sensores puede utilizarse en la programación de robots de muy diversas formas, entre las que se pueden destacar las siguientes:

- Modificar la trayectoria.
- Elegir entre diversas alternativas.
- Obtener la identidad y posición de objetos y sus características.
- Cumplir con restricciones externas.

8.2.5. Control del movimiento del robot

Indudablemente, un método de programación de robots debe incluir la posibilidad de especificar el movimiento del robot. Además del punto de destino, puede ser necesario especificar el tipo de trayectoria espacial que debe ser realizada, la velocidad media del recorrido o la precisión con que se debe alcanzar el punto destino. Incluso en ocasiones puede ser necesario indicar si el movimiento debe realizarse en cualquier caso o debe estar condicionado a algún tipo de circunstancia, procedente, por ejemplo, de la medida proporcianada por un sensor.

Tal y como se vio en el Capítulo 6 dedicado al control cinemático, las trayectorias de un robot pueden ser punto a punto, coordinadas o trayectoria continua, englobando este último caso a la línea recta, interpolación circular y otras.

En cuanto a la especificación de la velocidad, suele ser frecuente indicarla en la propia instrucción de movimiento como tanto por ciento de una velocidad base definida aparte. De este modo se facilita la alteración de la velocidad de todo el programa o de una parte de él sustituyendo una única instrucción.

En muchas ocasiones, el movimiento del robot en un entorno con obstáculos obliga a la utilización de una trayectoria en línea recta, para así asegurar que el robot no colisione con ninguno de ellos. No obstante, muchos sistemas de programación disponen de los denominados puntos de paso o *via points*, para resolver este tipo de situaciones. El modo en que una determinada configuración (posición y orientación) se define como punto de paso, está relacionado con la especificación de la precisión con que se desea que el robot alcance dicho punto.

El control (cinemático-dinámico) del robot, recibe las referencias de posición procedentes del programa, no admitiendo una nueva referencia hasta que el extremo del robot no alcanza la referencia en vigor con la precisión indicada. De este modo, si la precisión con la que se desea alcanzar una determinada configuración se define como *baja*, el extremo del robot se encaminará hacia la configuración siguiente sin haber llegado a la anterior y sin tener que disminuír apenas su velocidad. El resultado final es que el movimiento gana en continuidad y velocidad a costa de perder una precisión innecesaria en esos puntos de paso (Figura 8.6).

Se ha de tener en cuenta también la posibilidad de que el robot al recorrer la trayectoria generada tenga que pasar por un punto singular, ya sea porque los valores de sus ejes no estén definidos de forma biunívoca o porque alguno de ellos deba de alcanzar una velocidad excesiva para mantener la posición, orientación y velocidad deseadas del extremo. Algunos sistemas, como el RAPID [ABB-94], permiten especificar qué tipo de interpolación se debe utilizar de forma general durante la ejecución de una trayectoria cuando se ha de pasar por estos puntos singulares.

La consideración de las señales captadas por los sensores en la especificación de los movimientos del robot, puede hacerse a varios niveles. Una primera posibilidad responde a la interrupción del movimiento del robot por verificarse algún tipo de condición externa programada. Esta posibilidad ya

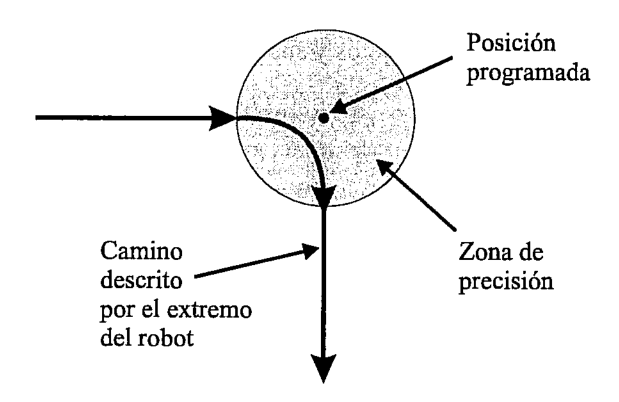


Figura 8.6. Funcionamiento de un punto de paso (*via point*).

comentada en el manejo de Entrada-Salida se suele denominar movimiento protegido o monitorizado (*guarded motions*).

La segunda alternativa, implica la modificación del movimiento, en cuanto a la situación de destino o velocidad, según la información captada del entorno. De este modo el movimiento del extremo del robot queda alterado, adaptándose a las necesidades de un entorno cambiante o parcialmente indeterminado. Estos movimientos se conocen como acomodaticios (*compliant motions*). En la práctica los movimientos monitorizados se implementan mediante el uso de interrupciones, mientra que los acomodaticios se realizan mediante el uso de funciones especiales (función ALTER en VAL II o V+).

8.2.6. Control del flujo de ejecución del programa

Al igual que en cualquier lenguaje de programación de propósito general, un lenguaje de programación de robots ha de permitir al programador especificar de alguna manera un flujo de ejecución de operaciones. Se emplean para ello las estructuras habituales de bucles (*for*, *repeat*, *while*, etc.). También es de especial importancia la capacidad de procesamiento en paralelo, tanto para el control de varios robots trabajando conjuntamente bajo las órdenes de un solo programa, como para el control de un único robot en una celda de trabajo cuyos equipos se encuentran igualmente bajo el control del programa del robot. Se emplea para ello señales de sincronismo basadas en semáforos y ejecución de tareas en paralelo.

Es importante para la mayoría de las aplicaciones el tener un control de interrupciones mediante las que distintos equipos en funcionamiento puedan interaccionar con el robot. Se debe poder fijar la prioridad en el tratamiento de las mismas, así como activarlas y desactivarlas durante la ejecución de diversas fases del programa.

8.3. EJEMPLO DE PROGRAMACIÓN DE UN ROBOT INDUSTRIAL

Para concretar de forma práctica como se realiza la programación textual de un robot industrial, se desarrolla a continuación un caso concreto utilizando un lenguaje textual de carácter experimental, como es el Código-R [FELIÚ-85]. Este lenguaje fue desarrollado para su utilización sobre el robot DISAM-E65. Se trata de un código de bajo nivel que sirve de enlace entre el controlador del robot y un lenguaje de más alto nivel, pudiendo ser escrito directamente o ser generado por otro sistema de programación superior, como es el caso del LRS [BARRIENTOS-86].

La aplicación a programar, mostrada en la Figura 8.7, consiste en una célula de verificación y empaquetado de componentes. Los componentes llegan a la célula a través de un alimentador, de donde el robot ha de cogerlos y realizar un test específico de validación sobre ellos. Si el test resulta positivo, se coloca el componente en el primer agujero libre del pallet de 3x5 situado en una cinta de transporte contigua. Esta cinta se mueve mediante un motor controlado por el propio robot. Si el test resultase negativo (componente defectuoso), se colocará el componente en una caja de componentes desechados.

El robot utilizado en la aplicación es de tipo SCARA, de cuatro grados de libertad: dos para posicionamiento en (x, y), uno para subir y bajar la pinza en z y el cuarto para girar la pinza sobre el eje z. El tercer eje tiene la particularidad de que está accionado neumáticamente, por lo que sólo posee dos posibles posiciones: arriba y abajo.

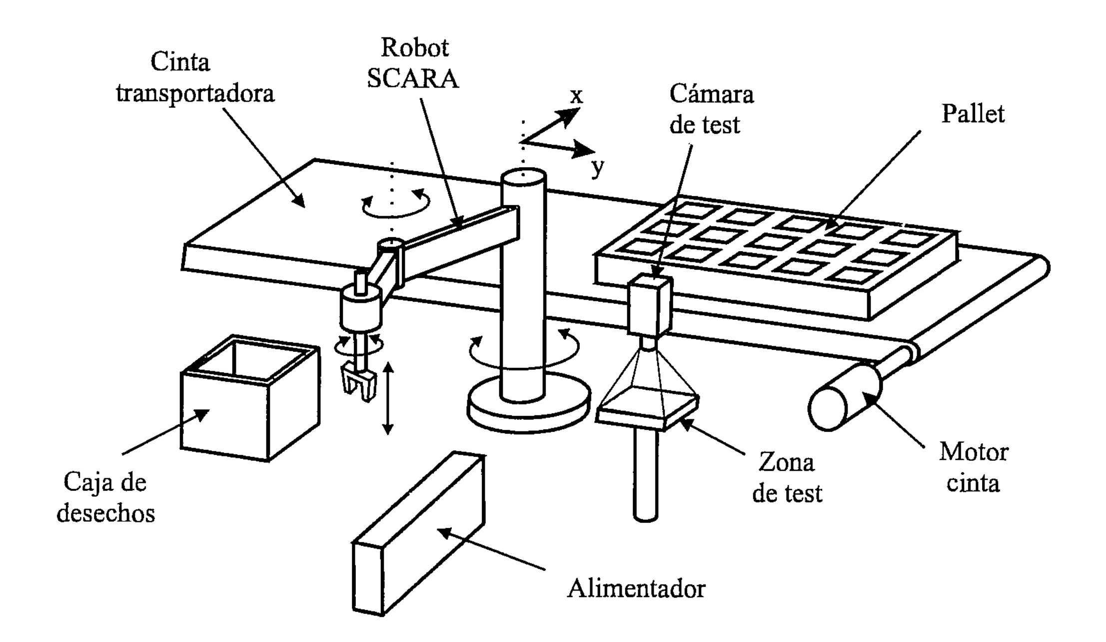


Figura 8.7. Esquema de la aplicación programada con Código-R.

Como señales de control de la célula por parte del sistema de control del robot se utilizan diversas señales digitales de entrada y salida. Como señales de entrada digitales se tienen:

Ea : Indica si hay componente en el alimentador.
Eh : Indica la existencia de componente en la herramienta (pinza).
Et : Indica si el test ha resultado positivo.
Ep : Indica si el pallet está disponible.

Y como salidas digitales:

Sn : Alarma que indica la necesidad de un nuevo componente.
Sh : Alarma que indica que no hay componente en la pinza.
Sm : Señal de control del motor de la cinta de transporte.
St : Señal de activación del test.

En la Tabla 8.1 se describen de forma breve el formato simplificado de las instrucciones del Código-R, necesarias para poder compreder sin dificultad el código del programa que se muestra a continuación. Existen además dos macroinstrucciones para coger la pieza (COG) y dejar la pieza (DEJ).

Tabla 8.1. Instrucciones del lenguaje Código-R

Nom.	Parámetros	Descripción	Ejemplo
POS$	Variable posición (x,y)	Posiciona el extremo del robot en la posición especificada.	POS$ Pa
SBR	Nº Subrutina	Llamada a una subrutina.	SBR 300
BUC	Nºrep.,identificador	Bucle de repetición.	BUC 5, 1
REP	Identificador de bucle	Fin de bucle.	REP 1
OPE	Var,Operacion,Var	Realización de una operación entre variables.	OPE Pv=Pa+Pi
SAL*	Nº de linea, condición	Salto condicional.	SAL* 100, Ea=1
SAL	Nº de línea	Salto incondicional.	SAL 104
EST	Valor númerico	Esperar un tiempo en segundos.	EST 2
GPZ	Variable orientación	Girar pinza a una orient.	GPZ Oa
ESE	Id. Entrada = valor	Esperar entrada.	ESE Ea=1
RET	-	Retorno de subrutina.	RET
VEL	tipo y velocidad	Asignar velocidad.	VEL a=2000
MON	Id entrada = valor, linea de salto, tiempo de espera	Monitoriza el valor de una señal un tiempo, si la señal toma el valor indicado salta a línea.	MON Ea=1,506,30
VAR	Id.variable = posición	Definición de var. de posición.	VARPa= 100,100
VARa	Id.variable = orientación	Definición de var. de orientación.	VARa Op = 100
SUB	-	Subir la pinza.	SUB
BAJ	-	Bajar la pinza.	BAJ
SLD	Id.Salida = valor,tiempo	Activar salida durante un tiempo. Por defecto es siempre.	SLD Sn=1,10

Macroinstrucciones

```
MAC    COG              ; Definición de la macroinstrucción coger
  PINZA = ABRIR
  BAJ
  EST  0.5
  PINZA = CERRAR
  SUBIR
END MAC

MAC  DEJ                ; Definición de la macroinstrucción DEJAR
  BAJ
  PINZA = ABRIR
  EST  0.5
  SUB
  PINZA = CERRAR
END MAC
```

El programa desarrollado para realizar esta aplicación consta de un módulo principal y varias subrutinas. A continuación se muestra el código de programa de cada una de ellas, precedidas de una breve explicación de su funcionamiento.

Inicialización de variables

Se definen en este bloque de datos inicial la velocidad lineal y angular del efector final, así como distintas variables con posiciones y orientaciones a utilizar durante el desarrollo de la aplicación.

```
VEL   =2000           ; Velocidad base de posicionamiento
VELa  =100            ; Velocidad de giro de la pinza

VAR   Pa = -450,275   ; Posición del alimentador
VAR   Pt =    0,450   ; Posición del dispositivo de test
VAR   Pi =  330,-30   ; Posición de comienzo del pallet
VAR   Pd = -250,450   ; Pos. almacén de comp. defectuosos


VAR   Pv =  330,-30   ; Pos. pallet libre y auxiliar
VAR   Pf =  330,-30   ; Pos. de la fila libre
VAR   Ic =   80,  0   ; Incremento de columna
VAR   If =    0, 80   ; Incremento de fila

VARa  Op = 100        ; Orientación del comp. en pallet
VARa  Oa =   0        ; Orientación del comp. en alimentación
VARa  Ot =  50        ; orientación de dispositivo de test
```

Proceso de llenado de pallet

Primero se pide un nuevo pallet, y después de coger el componente, se realiza el test. Si el test resulta positivo, se coloca el componente en el pallet y se actualiza la variable que representa la primera posición libre en el pallet. Si el test es negativo, se desecha el componente y se deja en la zona de componentes defectuosos. Si desaparace el pallet durante el llenado, se pedirá un nuevo pallet y se reiniciará el proceso.

```
100  SBR  300              ; Solicita pallet nuevo
101  BUC  5,filas          ; Bucle de indexación de filas
102    OPE  Pv=Pf          ; Comienzo de fila
103    BUC  3,columnas     ; Bucle de indexación de columnas
104      SAL* 100,Ep=0     ; No hay pallet; comenzar proceso
105      SBR  500          ; Tomar componente del alimentador
106      SBR  600          ; Realizar test
107      SAL* 110,Et=1     ; Componente válido. Situarle en pallet
108      SBR  700          ; Componente defectuoso. Rechazarle
109      SAL  104          ; Coger un nuevo componente
110      SLD  St=0         ; Test pasado, desactivar test y sacar
111      SUB               ; componente del dispositivo de test
112      SBR  400          ; Colocar componente en pallet
113      OPE  Pv=Pv+Ic     ; Incrementar posición de columna
114    REP columnas        ; fin de indexación de columnas
115    OPE Pf=Pf+If        ; Incrementar fila
116  REP  filas            ; Fin indexación filas, pallet lleno
117  SAL  100
```

Subrutina de nuevo pallet

El propósito de esta subrutina es garantizar que el pallet nuevo está posicionado en la posición de llenado, inicializando las variables Pv y Pf a la primera posición del pallet. Para conseguir esto, se activa la salida que controla el motor de movimiento de la cinta.

```
300  OPE  Pv=Pi       ; Inicialización de Pv
301  OPE  Pf=Pi       ; Inicialización de Pf
302  SLD  Sm=1        ; Pedir nuevo pallet, activar cinta
303  EST  2           ; Tiempo de evacuación del pallet anterior
304  ESE  Ep=1        ; Espera llegada de pallet nuevo
305  SLD  Sm=0        ; Parar cinta
306  RET              ; Fin de nuevo pallet
```

Subrutina de colocar en pallet

Esta rutina coloca, con la adecuada orientación de la pinza, un componente en la primera posición libre en el pallet.

```
400  POS$ Pv            ; Ir a posición libre en pallet
401  GPZ  Op            ; Con la correcta orientación
402  DEJ                ;
403  RET                ; fin de colocar en Pallet
```

Subrutina colocar componente

Espera durante un máximo de 30 segundos la llegada de un componente al alimentador. Si no llega, emite una alarma durante 10 segundos y lo vuelve a intentar. Si llega, lo recoge, verificando que está en la pinza del robot. Si no está (por ejemplo, porque ha resbalado) emite una alarma y trata de realizar la operación de nuevo.

```
500  POS$ Pa            ; Posicionarse sobre el alimentador
501  GPZ  Oa            ; Orientándose correctamente
502  MON  Ea=1,506,30   ; Monitoriza la entrada 1 durante 30s máx.
503  EST  30            ; Espera de 30 segundos
504  SLD  Sn=1,10       ; Si no llega componente en 30s
505  SAL  502           ; Alarma (Sn) durante 10s y repite acción
506  COG                ; Llega componente, cogerlo.
507  SAL* 510,Eh=0      ; Verificar que componente ha sido cogido
508  RET                ; Fin de tomar componente
510  SLD  Sh=1,5        ; Fallo al coger el componente
511  SAL  500           ; Emitir alarma (Sh) durante 5s y reintentar
```

Subrutina de test

Esta subrutina coloca el componente en el dispositivo de test y lo activa.

```
600  POS$ Pt            ; Posicionarse sobre dispositivo de test
601  GPZ  Ot            ; orientarse correctamente
602  BAJ                ; Introducir componente en dispositivo
603  SLD  St=1          ; Activar test
604  RET                ; Fin de test
```

Subrutina de rechazo de componente

Esta subrutina rechaza los componentes que no pasan el test, y los deja en el almacén de componentes defectuosos.

```
700  SLD  St=0          ; Componente defectuoso
701  SUB                ; Sacar componente del dispositivo de test
702  POS$ Pd            ; Posicionarse sobre almacen de piezas malas
703  DEJ                ; Dejar componente
705  RET                ; Fin de rechazar componente
```

8.4. CARACTERÍSTICAS BÁSICAS DE LOS LENGUAJES RAPID Y V+

En este apartado se realiza un breve repaso a las características de dos lenguajes de programación comerciales. Se trata de los sistemas de programación de robots V+ de Adept Technologies y RAPID de ABB. Con este repaso se pretende que el lector conozca de forma práctica la estructura real de un lenguaje de programación de robots. La información facilitada para ambos lenguajes ha sido extraída de los respectivos manuales [ABB-94] [ADEPT-89], y en ningún caso se pretende sustituirlos, tan sólo realizar una breve introducción a la programación de ambos sistemas. Para la correcta comprensión de las explicaciones es conveniente que el lector posea algunas nociones básicas de programación en lenguajes informáticos de programación convencionales.

8.4.1. El lenguaje de programación RAPID

El lenguaje RAPID, desarrollado por la empresa ABB para su nueva generación de robots, constituye uno de los últimos lenguajes de programación de robots aparecido en el mercado. Su lanzamiento en el año 1994 supone un nuevo paso adelante en el desarrollo de herramientas de programación para robots comerciales.

Constituye un lenguaje de programación textual de alto nivel que incluye algunas características importantes, tales como la utilización de funciones y procedimientos, la posibilidad de usar rutinas parametrizables, la estructura completamente modular del programa o la posibilidad de declarar rutinas y datos como locales o globales.

Un programa escrito en RAPID consiste en una serie de instrucciones que describen el trabajo del robot. Cada instrucción tiene asociada una serie de parámetros que definen de forma completa la acción a realizar. Estos parámetros específicos a cada instrucción pueden ser valores numéricos, referencias a un dato, expresiones de datos, llamadas a una funciones e incluso en algunos casos cadenas de caracteres.

Los programas se desarrollan en la paleta de programación bajo un entorno de menús dirigidos. A través del joystick situado en la propia paleta y de los diversos botones adicionales (tipo de movimiento, ejes externos, entradas/salidas) se puede también realizar entrada de datos por guiado e interaccionar con el entorno en cualquier momento. Los programas también pueden ser escritos directamente en un PC con un editor de texto convencional, para ser posteriormente volcados en la memoria del robot a través de un disquete.

Se trata de un lenguaje altamente estructurado que recuerda a un lenguaje de programación de propósito general. Los programas desarrollados en RAPID se denominan tareas e incluyen el programa en sí junto con varios módulos de sistema, módulos que contienen rutinas y datos de tipo general, independientes del programa pero que pueden ser utilizados por él en cualquier momento. A su vez, el programa puede ser dividido en varios módulos, uno de los cuales ha de ser el principal. Cada uno de estos módulos contiene submódulos de datos, además de diversas rutinas de ejecución. La Figura 8.8 muestra de forma esquemática la organización de esta estructura. La definición de módulos se realiza

especificando su nombre y sus atributos, como por ejemplo que se trata de un módulo del sistema (*SYSMODE*) o que no se puede modificar (*VIEWONLY*).

Existen tres tipos diferentes de rutinas o subprogramas posibles:

- **Procedimiento**: Rutina que no devuelve ningún valor y se utiliza como una instrucción.
- **Función**: Rutina que devuelve un dato de tipo específico y que se utiliza como una expresión.
- **Rutina TRAP**: Son rutinas que se asocian a interrupciones y se ejecutan cuando éstas se activan. No pueden llamarse nunca de forma explícita.

Los datos a manejar pueden ser de dos tipos:

- **Tipo de dato atómico**: No se define en función de otro tipo y no se puede dividir en diferentes componentes.
- **Tipo de dato registro**: Está compuesto por una serie ordenada de componentes con sus respectivos nombres. Los componentes pueden ser a su vez de tipo atómico o de tipo registro.

Estos datos, que se pueden definir como globales o locales tanto en módulos como en subrutinas, pueden a su vez ser definidos como:

- **Constantes**: (*CONS*) representan datos de un valor fijo a los que no se puede reasignar un nuevo valor.
- **Variables**: (*VAR*) son datos a los que se les puede asignar un nuevo valor durante la ejecución del programa.

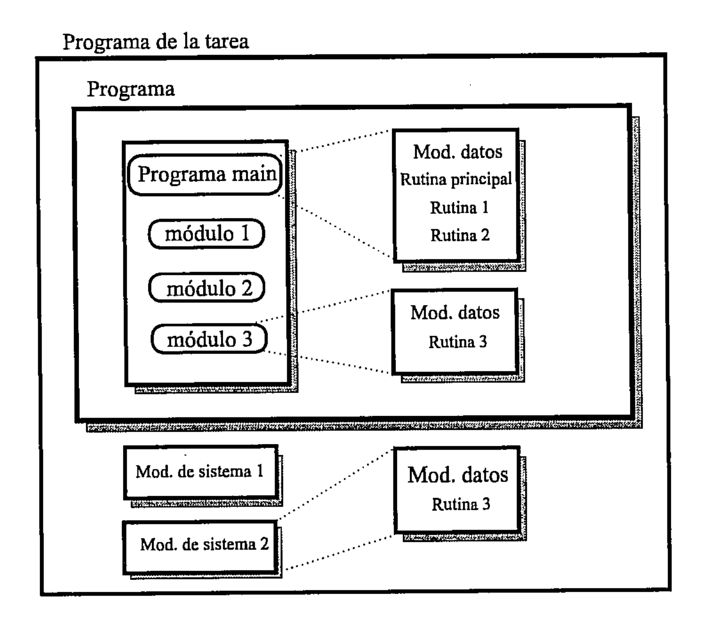


Figura 8.8. Estructura de un programa en RAPID.

- **Persistentes**:(*PERS*) se trata de variables en las que cada vez que se cambia su valor durante la ejecución del programa, también se cambia el valor de su inicialización.

Los tres tipos básicos de datos habituales con los que se trabaja son:

- Numérico: *num.*
- Lógico: *bool.*
- Cadena: *string.*

Es importante destacar la existencia de una serie de estructuras predefinidas de datos con distintos campos para la utilización por parte del usuario. Se muestran a continuación algunos ejemplos:

- **Confdata**

Descripción: estructura para especificar la configuración del robot.
Campos: *cf1*: cuadrante del eje 1
cf4: cuadrante del eje 4
cf6: cuadrante del eje 6
Ejemplo:

VAR confdata conf10:=[1,-1,0]

- **Loaddata**

Descripción: describe la carga colocada en la muñeca del robot.
Campos: *mass*: peso de la carga en kilogramos
cog: centro de gravedad de la carga
aom: orientación de los ejes de inercia en el centro de gravedad expresada como cuaternios.
ix, iy iz: momentos de inercia de la carga alrededor del eje **x**, eje **y** y eje **z** expresado en kgm^2.

Ejemplo:

PERS loaddata pieza:=[5,[50,0,50],[1,0,0,0],0,0,0];

- **Tooldata**

Descripción: estructura para especificar las características de una herramienta.
Campos: *robhold*: define si el robot tiene la herramienta o no.
tframe: sistema de coordenadas de la herramienta (posición y orientación de la herramienta)
tload: dato tipo *loaddata.*
Ejemplo:

PERS tooldata pinza:=[TRUE,[[97,0,220],[0.924,0,0.383,0]]
[5,[-23,0,75],[1,0,0,0],0,0,0]]

- **Robtarget**

Descripción: se utiliza para definir la localización del robot y de los ejes externos
Campos: *trans*: desplazamiento en x,y,z del sistema de coordenadas
rot: rotación del sistema de coordenadas como cuaternios
robconf: configuración del robot (cf1,cf4,cf6 y cfx)
extax: posición de los ejes externos
Ejemplo:
VAR robtarget posicion:= [[600,500,225],[1,0,0,0],[1,1,0,0], [11,12,9E9,9E9,9E9,9E9]];

- **Motsetdata**

Descripción: se utiliza para la definición de algunos parámetros de movimiento que afectan a las instrucciones de posicionamiento del programa.
Campos: *vel.oride*: velocidad como porcentaje de la programada
vel.max: velocidad máxima en mm/s
sing.wrist: desviación de la orientación de la herramienta para evitar singularidad de la muñeca.
sing.arm: desviación de la orientación de la herramienta para evitar singularidad en el brazo.
...
conf.ax1:desviación máxima del eje 1
conf.ax4:desviación máxima del eje 4
conf.ax6:desviación máxima del eje 6
...

El sistema de programación RAPID posee un variada gama de instrucciones para controlar el flujo de ejecución del programa, de entre las que destacan: llamadas a rutinas con distintos parámetros, instrucciones de repetición tipo *FOR*, tipo *IF* o tipo *WHILE*. Se cuenta además con las operaciones aritméticas habituales (asignación,+,-,*,/) que se pueden asociar tanto a números como a vectores o matrices (con las limitaciones oportunas), con algunas funciones matemáticas particulares como son: *Add, Clear, Incr, Decr, Abs* y operadores booleanos para el tratamiento de variables tipo *bool*.

El control de movimientos del robot se encuentra especialmente cuidado, existiendo primeramente unos parámetros generales de movimiento que afectan a todas las instrucciones de movimiento, a no ser que en la propia instrucción se indique lo contrario. Alguno de estos parámetros son:

- *AccSet*: máxima aceleración permitida.
- *SingArea*: método de interpolación en puntos singulares.
- *GripLoad*: peso de la pieza a manipular.
- *VelSet*: velocidad máxima y sobrevelocidad.

Las intrucciones generales de movimiento se definen en base a cómo posicionar el punto central de la herramienta (TCP), pudiéndose utilizar las siguientes.

Tabla 8.2. Juego de instrucciones del RAPID

:=	Asignar un valor
Abs()	Obtener el valor absoluto
AInput()	Leer el valor de una señal de entrada analógica
AccSet	Reducir la aceleración
Add	Sumar un valor numérico
Clear	Borrar un valor
ClkStart	Iniciar un reloj para la toma de tiempos
ClkStop	Parar un reloj para la toma de tiempos
comment	Comentario
CompactIF	Si se cumple una condición, entonces... (una instrucción)
ConfJ	Controlar la configuración durante movimiento articular
ConfL	Monitoriza la configuración del robot durante movimiento en línea recta
Decr	Decrementar en 1
EXIT	Terminar la ejecución del programa
FOR	Repetir un número de veces
GetTime()	Leer el valor de la hora actual como valor numérico
GOTO	Ir a una nueva instrucción
GripLoad	Definir la carga del robot
HoldMove	Interrumpir el movimiento del robot
IF	Si se cumple una ocndición, entonces...; de otra manera...
Incr	Incrementar en 1
InvertDO	Invertir el valor de una salida digital
label	Nombre de una línea
LimConfL	Definir la desviación permitida en la configuración del robot
MoveC	Mover el robot en movimiento circular
MoveJ	Movimiento articular del robot
MoveL	Movimiento del robot en línea recta
Offs()	Desplazamiento de la posición del robot
Open	Apertura de un fichero o de un canal serie
Present()	Comprobar que se utiliza un parámetro opcional
ProcCall	Llamada a un nuevo procedimiento
PulseDO	Generar un pulso en una señal digital de salida
RAISE	Llamada a un manejador de errores
RelMove	Continuar con el movimiento del robot
Reset	Reset de una salida digital
RETRY	Recomenzar tras un error
RETURN	Terminar la ejecución de una rutina
Set	Set de una salida digital
SetAO	Cambiar el valor de una salida analógica
SetDO	Cambiar el valor de una salida digital
SetGO	Cambiar el valor de un grupo de salidas digitales
SingArea	Definición de la interpolación alrededor de puntos singulares
Stop	Parar la ejecución de un programa
TEST	Dependiendo del valor de la expresión...
TPErase	Borrar el texto de la paleta de programación
TPReadFK()	Leer las teclas de función de la paleta de programación
TPWrite	Escribir en la paleta de programación
VelSet	Cambiar la velocidad programada
WaitDI	Esperar hasta el set de una entrada digital
WaitTime	Esperar un tiempo determinado
WaitUntil	Esperar hasta que se cumpla una condición
WHILE	Repetir mientras ...
Write	Escribir en un fichero de caracteres o en un canal serie
WriteBin	Escribir en un canal serie binario

- *MoveC*: mover el TCP de manera que éste describa un círculo.
- *MoveJ*: mover el robot con trayectoria articular (isocrona o coordinada en articulares).
- *MoveL*: mover el TCP del robot describiendo una línea recta.

Estas instruciones tienen asociados una serie de parámetros, definidos como variables de los tipos generales predefinidos, algunos de los cuales ya se han visto. La forma de programar estas instrucciones es:

MoveJ [\Conc] ToPoint Speed [\V] | [\T] Zone [\Z] Tool [\WObj]

- *[\Conc]*: ejecución concurrente. Se ejecutan las siguientes instrucciones a medida que el robot se mueve.
- *ToPoint*: punto destino del robot (tipo de dato: *robtarget*). Si se graba directamente por guiado aparecerá un asterisco (*).
- *Speed*: velocidad que afecta al movimiento (tipo de dato: *speeddata*).
- *[\V]*: velocidad específica del TCP (tipo de dato: *num)*.
- *[\T]*: tiempo en segundos que debe tardar en realizar el movimiento (tipo de dato: *num*).
- *Zone:* zona del movimiento para que se empiece a ejecutar la siguiente instrucción (tipo de dato: *zonedata*).
- *[\Z]:* precisión del TCP (tipo de dato*: num*).
- *Tool*: la herramienta en uso (tipo de dato: *tooldata*).
- *[\WObj]*: objeto de trabajo con el que el movimiento está relacionado (tipo de dato: *wobjdata*).

Los movimientos se programan a base de posiciones definidas, es decir, se le dice al robot que se mueva desde donde se encuentra a una posición determinada.

El sistema permite manipular señales de entrada/salida de diversas maneras. Se definen los nombres de las señales en los parámetros del sistema y a través de una llamada a la función correspondiente se puede leer su valor (*DInput*) o modificarlo (*SetDO*). Es posible, igualmente, la manipulación al mismo tiempo de un grupo de señales.

Se permite también la comunicación con la paleta de control, tanto para mostrar mensajes en pantalla (*TPWrite*), como para leer la información que se introduzca a través de ella (*TPRedStr*). Es posible la lectura/escritura a través de un canal o fichero tanto en ASCII como en binario. Existe además la posibilidad de que el programador controle las acciones a realizar cuando en una determinada subrutina aparece un error no previsto, por ejemplo división por cero.

Son importantes también las instrucciones que posibilitan el manejo de un reloj de tiempo, que se puede utilizar durante la ejecución del programa para controlar determinadas acciones.

En la Tabla 8.2 se muestra el juego de instrucciones del RAPID con una breve explicación de la aplicación de cada una de ellas.

Programa ejemplo

Se presenta a continuación un programa ejemplo de una operación sencilla de manipulación. El robot retira de una cinta transportadora aquellas piezas identificadas como defectuosas. (Figura 8.9.) La operación se desarrolla como sigue: el robot se encuentra en espera hasta la llegada de una señal indicando la existencia de una pieza defectuosa sobre la cinta transportadora. El robot procede entonces

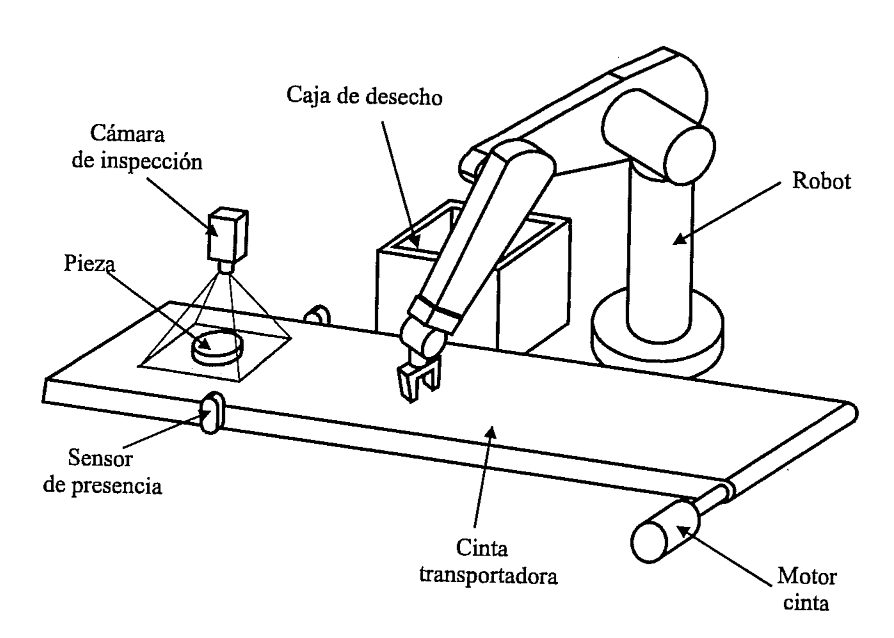


Figura 8.9. Célula robotizada de inspección para ejemplo de programación.

a parar la cinta y a coger la pieza y a depositarla en un almacén de piezas defectuosas. El propio robot se encarga de activar de nuevo el movimiento de la cinta una vez la pieza ha sido cogida. Tras la operación, el robot vuelve a su posición inicial y se repite de nuevo el ciclo.

El programa cuenta con una rutina principal junto con varias subrutinas específicas, además de la definición de las variables correspondientes.

Definición de variables

El robot posee en su extremo una pinza para la manipulación de piezas, definida dentro del programa mediante la variable **herramienta** de tipo *tooldata.* A su vez se ha definido una variable *loadata* llamada **carga** para definir la carga a transportar por la pinza.

PERS tooldata **herramienta** := [FALSE,[[97,0,223],[0.924,0,0.383,0]],
[5,[-23,0,75],[1,0,0,0],0,0,0]]
PERS loaddata **carga** := [5,[50,0,50],[1,0,0,0],0,0,0];

A las entradas y salidas digitales se les puede asignar identificadores alfanuméricos. Esta asiganción es válida para todos los programas, debiéndose realizar en la etapa de definición de parámetros del sistema. Posteriormente, los programas que vayan a usar estas variables, deberán declararlas con la instrucción *VAR*. La salida correspondiente a la activación de la pinza se ha denominado **pinza**. La entrada digital correspondiente a la detección de una pieza defectuosa se llama **pieza_defectuosa**. La

salida correspondiente a la activación del motor de la cinta se llama **activar_cinta**. Existe además una entrada digital llamada **terminar** para indicar que la aplicación ha terminado.

```
VAR signaldo pinza                ;señal de activación de pinza
VAR signaldo activar_cinta        ;señal de activación de cinta
VAR signaldi pieza_defectuosa     ;señal de pieza defectuosa
VAR signaldi terminar             ;señal de terminar programa
```

Es necesario definir una configuración inicial en la que el robot espera la señal que le indica que puede recoger la pieza defectuosa. Ésta se denomina **conf_espera** y tiene estructura de *robtarget*.

```
VAR robtarget  conf_espera := [[600,500,225],[1,0,0,0],[1,1,0,0],
                              [9E9,9E9,9E9,9E9,9E9,9E9] ];
```

Existen a su vez una serie de variables que se utilizan durante la ejecución del programa y que ya están predefinidas. Se incluyen entre ellas los parámetros de precisión (z20,z30,FINE...) y los de velocidad (VMAX,V1000,V800...) pertenecientes a la instrucción de posicionamiento, además de una variable (LOAD0) que indica la ausencia de carga en la pinza.

Rutinas de control de la pinza

Para el control directo de la pinza se han definido dos procedimientos, a los que se puede llamar en cualquier momento dentro del programa principal o desde cualquier otra rutina.

```
PROC Coger()
  Set pinza              ; Cerrar la pinza activando la señal digital pinza
  WaitTime 0.3           ; Esperar 0.3 segundos
  GripLoad  carga        ; Señalar que la pieza está cogida
ENDPROC

PROC Dejar()
  Reset pinza            ; Abrir la pinza
  WaitTime 0.3           ; Esperar 0.3 segundos
  GripLoad LOAD0         ; Señalar que no hay pieza cogida
ENDPROC
```

Rutina de coger la pieza de la cinta

Se coge una pieza de la cinta transportadora. Se realiza primero una aproximación en coordenadas angulares, movimiento coordinado en articulares (trayectoria isocrona), para más tarde acercarse en línea recta y con precisión a coger la pieza.

```
PROC  Coger_pieza()
  MOVEJ*,  VMAX, z60, herramienta         ; Mov. en articulares con poca precisión
  MOVEL*, V500, z20, herramienta          ; Mov. línea recta con precisión
  MOVEL*, V150, FINE, herramienta         ; Bajar con precisión máxima
  Coger                                   ; Coger la pieza
  MOVEL*, V200, z20, herramienta          ; Subir con la pieza cogida
ENDPROC
```

Rutina de dejar la pieza

El robot deja la pieza sobre el almacén de piezas defectuosas

```
PROC  Dejar_pieza()
  MOVEJ*, VMAX, z30, herramienta          ; Mover hacia almacén piezas malas
  MOVEJ*, V300, z30, herramienta
  Dejar                                   ; Dejar la pieza.
ENDPROC
```

Rutina de ir a la posición de espera

Se mueve al robot desde la posición en la que se encuentre hasta la posición de espera.

```
PROC Ir_posición_espera()
  MoveJ conf_espera, VMAX, z30, herramienta      ; Mover a posición inicial
ENDPROC
```

Programa principal

El programa principal comprende todas las instrucciones a ejecutarse para la realización práctica de la aplicación. Algunas son instrucciones básicas, mientras otras corresponden a llamadas a procedimientos. Primero se va a la posición inicial de espera y se aguarda a la indicación de que existe una pieza defectuosa. Tras parar la cinta y coger la pieza, el robot lleva la pieza al almacén de desechos y la deposita allí. Este proceso se repite hasta que se presione un botón asociado a la entrada digital **terminar**.

```
PROC main()
  Ir_posicion_espera;                       ; Mover a posición de espera
  WHILE DInput(terminar) = 0 DO             ; Esperar señal de terminar
   IF DInput(pieza_defectuosa) = 1 THEN     ; Esperar señal de pieza defectuosa
     SetDO activar_cinta,0;                 ; Parar cinta
```

```
      Coger_pieza                    ; Coger la pieza
      SetDO activar_cinta,1;         ; Activar señal de cinta
      Dejar_pieza                    ; Dejar la pieza
      Ir_posicion_espera             ; Ir a posición de espera
    ENDIF
  ENDWHILE
ENDPROC
```

Al sencillo programa presentado se le pueden añadir sucesivos refinamientos. Por ejemplo, se puede contemplar el caso de tener controlado mediante una señal digital que la pieza se encuentre agarrada por la pinza y que no se haya soltado debido a cualquier circunstancia accidental.

Para ello se puede utilizar una señal digital asociada a una interrupción, que se activará durante el transporte de la pieza. Si se detecta que la pieza se ha soltado, la interrupción llamará a una rutina tipo *TRAP* que parará de inmediato el movimiento del robot y presentará un mensaje por la paleta de programación indicando el accidente. Una vez arreglado el problema, la confirmación del operario permitirá al robot volver a la posición de espera.

Se define primero la variable de interrupción:

```
VAR intnum senal_pieza;
```

Dentro de la rutina principal main se asocia a la variable **senal_pieza** la rutina de interrupción correspondiente y se definen las variables:

```
ISignalDI senal_pieza                ; Asignar señal de interrupción
CONNECT senal_pieza TO etrap         ; Asignar rutina de interrupción
```

La interrupción deberá activarse únicamente cuando se transporte una pieza, luego se activará al coger la pieza y se desactivará al dejarla. De esta forma las rutinas coger y dejar quedan:

```
PROC Coger()
  Set pinza                 ; Cerrar la pinza activando la señal digital pinza
  WaitTime 0.3              ; Esperar 0.3 segundos
  GripLoad carga            ; Señalar que la pieza está cogida
  IWatch senal_pinza        ; Activar interrupción
ENDPROC

PROC Dejar()
  ISleep senal_pinza        ; Desactivar interrupción
  Reset pinza               ; Abrir la pinza
  WaitTime 0.3              ; Esperar 0.3 segundos
  GripLoad LOAD0            ; Señalar que no hay pieza cogida
ENDPROC
```

Por último, se deberá incluir la rutina TRAP de ejecución cuando se da la señal de interrupción.

```
TRAP etrap
 HoldMove                                   ; Para la ejecución del robot
 TPErase                                    ; Borrar pantalla de paleta
 TPWrite "ERROR: La pieza se ha soltado de la pinza"
 TPWrite ""
 reg:= TPReadFK ("Cuando el sistema esté listo pulse Start",\FK1:="Start");
 RelMove                                    ; Reanudar el movimiento
 Ir_posicion_espera                         ; Ir a posición inicial
ENTRAP
```

Otra opción de mejora posible podría ser la inclusión de algún tipo de control sobre el paso en las cercanías de puntos singulares. Los puntos singulares resultan un problema importante de ejecución real de movimientos lineales por parte del extremo del robot. En un robot ABB articular de 6 gdl existen dos tipos posibles de puntos singulares: de muñeca (ejes 4 y 6 paralelos) y de brazo (centro de muñeca coincide con eje 1).

En el programa que se está mostrando sería conveniente añadir una instrucción para indicar que en las cercanías de un punto singular de muñeca se le permite al robot separarse ligeramente de la orientación establecida para así evitar la singularidad. Para ello se utilizaría la siguiente instrucción al principio del programa:

```
SingArea \Wrist;
```

Por último, otra posible mejora del programa consistiría en la ejecución concurrente de las instrucciones de movimiento junto con posibles instrucciones lógicas posteriores. Por defecto, durante la ejecución de una instrucción de movimiento el programa se queda bloqueado, sin que se ejecuten las instrucciones siguientes. Pero podría ser interesante permitir que mientras el robot va a la posición de espera, estuviera pendiente de las señales de terminación de la aplicación o de la presencia de una pieza defectuosa en la cinta. Para ello, en la instrucción de movimiento a la posición de espera se añade el parámetro \Conc, que indica la concurrencia, ejecutándose todas aquellas instrucciones lógicas que suceden a la instrucción de movimiento mientras se realiza éste.

```
MoveJ \Conc conf_espera, VMAX, z30, herramienta          ; mover a espera
```

8.4.2. El lenguaje de programación V+

V+ es un lenguaje de programación textual de alto nivel, desarrollado en 1989 por Adept Technology. Actualmente se utiliza en los robots Adept y en algunos robots de la firma Staübli, como por ejemplo en el modelo Rx 90. Durante la década de los setenta los robots se utilizaron de forma satisfactoria en tareas como soldadura o pintura, utilizando el guiado como método de programación, ya que este tipo de tareas no requieren interacción entre el robot y el entorno. En la actualidad, la utilización de los robots para la realización de tareas cada vez más complejas requiere la programación en lenguajes de alto nivel,

ya que es necesario utilizar la realimentación de los sensores, y este tipo de interacción sólo se puede realizar satisfactoriamente con métodos de programación que contengan condiciones.

Los lenguajes de programación de alto nivel, y en concreto V+, proporciona una mayor interacción entre el hombre y el robot. Además presenta claras ventajas como son:

- **Inteligibilidad**: Se puede realizar una buena documentación, así como un diseño ordenado y coherente del programa.
- **Fiabilidad**: Sobre todo en sistemas que deban responder a situaciones imprevistas.
- **Adaptabilidad**: Los programas permiten modificaciones, mejoras y ampliaciones con, relativamente, poco esfuerzo.
- **Transportabilidad**: Los programas se pueden desarrollar en una computadora (un PC, por ejemplo) para poder ser posteriormente introducidos en el controlador del sistema a través de un disquete. V+ cuenta con el protocolo KERMIT y el programa de aplicación XMODEM que permiten la transferencia de programas entre el controlador Adept y otro computador a través de una línea serie.

Un programa en V+ consiste en un conjunto de instrucciones. Cada línea o *step* se interpreta como una instrucción del programa. El formato general de cada línea es

número_línea *etiqueta* *instrucción* *;comentario*

A cada línea de programa se le asigna automáticamente un número de línea, que se ajusta automáticamente cuando se insertan o borran líneas. A medida que el programa evoluciona, los números de línea cambian, por lo que para identificar a una línea se le puede colocar una etiqueta, que no es más que un número entre 0 y 65535 colocado al principio de la línea. A continuación de la etiqueta (opcional) se coloca cualquier instrucción válida de V+. Por último se puede incluir en la línea un comentario que consiste en una frase escrita después del carácter ';'. El editor de programa de V+ comprueba automáticamente que la sintaxis es correcta cada vez que se introduce una nueva línea.

La primera línea de un programa en V+ es la instrucción ".PROGRAM", seguida del nombre del programa y de los parámetros que deba recibir o devolver. Tiene el formato:

.PROGRAM nombre(lista de parámetros)

donde el nombre es obligatorio y la lista de parámetros opcional. El final del programa se indica con una línea que contiene la instrucción ".END".

En el sistema V+ se pueden usar variables que, dependiendo de la forma en que acceden a ellas los programas, se pueden clasificar en tres tipos diferentes:

- **Globales**: Todos los programas que se encuentran en memoria tienen acceso a ellas. El principal problema que presentan es que hay que ser muy cuidadosos para que una subrutina u otro programa no modifiquen el valor que está usando un programa concreto. Para evitar este inconveniente se utilizan las variables locales y automáticas.
- **Locales**: Una variable se puede hacer local utilizando la instrucción *LOCAL*. Una variable local se crea la primera vez que se utiliza la instrucción *LOCAL*, y mantiene su valor entre llamadas

al programa. Es decir, si un programa crea una variable local y luego se sale del programa, la variable mantiene su valor hasta la siguiente vez que se entra al programa. Sin embargo, si diferentes tareas usan el mismo programa, o si se usa recursivamente por una tarea, pueden existir conflictos con los valores que toma la variable en cada caso. Para evitar este inconveniente se usan las variables automáticas. Otra desventaja de las variables locales es que continúan ocupando memoria una vez que el programa al que están asociadas termina su ejecución.

- **Automáticas**: Las variables automáticas se crean dentro de un programa con la instrucción *AUTO*. Son parecidas a las variables locales ya que sólo se puede acceder a ellas desde el programa en que han sido creadas. Sin embargo, la diferencia estriba en que, cada vez que se entra al programa, se crea una copia separada de cada variable automática, y el valor de dicha copia se pierde cuando se sale del programa. Si un programa se usa por varias tareas a la vez, los valores de las variables automáticas son independientes en cada momento, con lo que se evitan los conflictos. Cuando se sale del programa, la memoria se libera automáticamente, con lo que los programas que no están activos al mismo tiempo pueden compartir la memoria usada por las variables automáticas. Aunque no se pueden considerar como un tipo independiente de variable, los argumentos de las subrutinas son un caso especial de variables automáticas. El alcance de los argumentos de las subrutinas es el mismo que el de las variables automáticas.

Es preciso señalar que el sistema V+ permite especificar el contexto de las variables locales y automáticas que se utilizan mediante comandos como HERE, POINT, TEACH, etc. La sintaxis general es la siguiente:

comando @tarea:programa parámetros

donde *comando* representa el nombre de un comando, y *parámetros* representa los parámetros normales del comando. El elemento opcional *@tarea:programa* especifica el contexto para las variables referenciadas en los parámetros del comando. *tarea* es un entero que indica una de las tareas del sistema y *programa* el nombre de un programa en la memoria del sistema.

Dependiendo del tipo de instrucciones o comandos que se utilicen, y de las acciones que pueden llevar a cabo, existen tres tipos diferentes de programas que se pueden realizar con el sistema V+ :

- **Programas de control del robot**: Controlan directamente el robot. Pueden contener cualquier tipo de instrucción, incluyendo instrucciones de movimiento del robot. Normalmente se ejecutan con la tarea principal (número 0), pero pueden ser ejecutados con cualquier tarea del sistema.
- **Programas de control de propósito general**: Es un programa que no controla al robot. Pueden existir uno o más programas de este tipo ejecutándose al mismo tiempo que un programa de control. Por ejemplo, se pueden usar estos programas para controlar procesos externos a través de líneas binarias externas. Se pueden comunicar con el programa de control mediante variables compartidas y señales software. En general, estos programas no pueden ejecutar directamente instrucciones que afecten al movimiento del robot. Sin embargo, y aunque no es habitual, un programa puede cambiar su clasificación durante la ejecución, pasando de ser un programa de propósito general a un programa de control y viceversa.

- **Programas de comandos del monitor:** Se componen de comandos de monitor más que de instrucciones de programa. Se usan para realizar secuencias de comandos del monitor que se usan frecuentemente.

Una de las características más importantes del sistema V+ consiste en la posibilidad de ejecución de diferentes programas al mismo tiempo. Así pues, se puede ejecutar, por ejemplo, un programa de control y otros programas adicionales. El sistema administra la ejecución de cada programa como una tarea independiente. El sistema V+ cuenta con siete tareas disponibles. El modo de ejecución del programa depende de la tarea que se use y del tipo del programa a ejecutar. De las tareas existentes, la número 0 es la más significativa dado que se usa generalmente para la ejecución del programa de control del robot, y es por tanto la tarea de más alta prioridad.

Las instrucciones de un programa se ejecutan normalmente de manera secuencial, aunque existen instrucciones como GOTO, CALL, IF...GOTO, WAIT, STOP, ETC. que pueden variar el flujo de ejecución del programa.

Una característica particular y muy potente de V+ es el proceso asíncrono, que consiste en la capacidad del sistema para responder a eventos (señales externas o errores) cuando estos ocurren. Si está habilitado el manejo de eventos, cuando alguno de estos ocurre, se produce la ejecución de un programa específico, llamado rutina de reacción, como si de una instrucción CALL se tratara. Se llama Proceso Asíncrono porque la ejecución no está sincronizada con el flujo normal del programa.

Los procesos asíncronos se habilitan con las instrucciones REACT, REACTE y REACTI. Cada una de las tareas usa estas instrucciones para preparar el procesado independiente de los eventos. Una rutina de reacción se ejecuta solamente si la prioridad del programa es menor que la de la reacción. El programa principal (normalmente el programa de control) se ejecuta habitualmente con prioridad cero, mientras que la menor prioridad que puede tener una reacción es uno. De esta forma, la reacción puede interrumpir normalmente el programa principal. Cuando se ejecuta una rutina de reacción, la prioridad del programa se ajusta automáticamente a la prioridad de la reacción para evitar que sea interrumpido por reacciones de igual o menor prioridad. Al salir de la rutina de reacción, el programa principal recupera la prioridad que tenía anteriormente.

Programa ejemplo

A continuación se presenta un programa de ejemplo en el que el robot ha de realizar operaciones de manipulación. En el programa se introduce un ejemplo del concepto de ejecución concurrente (ejecución de varios programas en paralelo) y de la habilitación de un proceso asíncrono para la gestión de errores. El robot ha de realizar una tarea de *pick and place* entre posiciones variables calculadas en un ordenador que se encarga de recibir y transmitir mensajes al robot. Los dos programas, el de comunicaciones con la estación y el de control del robot se ejecutan en paralelo.

El programa de comunicación tendrá el siguiente aspecto:

```
.PROGRAM comunica()

;PROGRAMA DE COMUNICACION CON LA ESTACION
lu_est = 10             ; asigna a la unidad lógica de la estación el valor 10
hay_cod_fun = FALSE     ; indica que no hay código de función disponible
$mens_ttir = ""         ; el mensaje a transmitir es una cadena vacía
```

```
hay_syst_err = FALSE          ; indica que no hay error del sistema

;CODIGOS DE ERROR
er_rob.no.ok = 256


;INICIALIZA EL CONTROL DEL ROBOT
SPEED 20                      ; se asigna la velocidad del robot
READY                         ; prepara al robot posicionándole en la posición de espera
DETACH (0)                    ; se libera al robot de la tarea #0
EXECUTE 1 robot()             ; se ejecuta el programa de control del robot con la tarea #1
                              ; a partir de aquí los dos programas se ejecutan en paralelo

;ESTABLECIMIENTO DE LA CONEXIÓN ROBOT-ORDENADOR
10  TYPE " Esperando establecer la conexión ..." ;mensaje por pantalla
    ATTACH (lu_est,0)              ; se asigna la línea de comunicación al programa
    ;se establece un bucle que controla si la comunicación se realiza con éxito
    IF IOSTAT (lu_est,0) <= 0 THEN
            TYPE "Error al intentar establecer la comunicación !!!!"
            DETACH (lu_est)        ;libera la unidad lógica de la tarea
            GOTO 10
    ELSE
            TYPE " Robot conectado a la estación"
    END

;RECEPCION Y DECODIFICACION DE ORDENES
```

;se establece un bucle que comprueba el estado del canal de comunicación y lee el mensaje mandado por el ordenador. Si hay un error del sistema se envía un mensaje través del canal lógico y se lee el nuevo mensaje.

```
20  DO
            IF (hay_syst_err) THEN
                    WRITE (lu_est) $mens_ttir, /S
            END
            READ (lu_est, , 1) $mens_rbdo
    UNTIL (IOSTAT(lu_est,0) <> -256)

    ;se comprueba el estado del canal de comunicación comprobando que la recepción es correcta
    IF IOSTAT(lu_est,0) <= 0 THEN
            TYPE "iostat error!!!", IOSTAT(lu_est,0)
            TYPE "Recepción incorrecta de orden"
            DETACH (lu_est)        ;se libera la unidad lógica de la tarea asignada
            GOTO 10
    END
```

```
        ;se asigna el código de la función recibida
        cod_fun = ASC($MID($mens_rbdo,1,1))

    ;EJECUCION DE ORDENES
    ;se comprueba si el robot está listo. Si lo está' se da paso al programa de control, si no, se envía un
    mensaje diciendo que el robot no está preparado
        IF ((STATE(1) <> 2) OR (cod_fun == 4)) THEN
                hay_cod_fun = TRUE   ;da paso al programa del robot que estaba en espera activa.
                WAIT (NOT hay_con_fun)        ;espera a que se haya ejecutado la orden
        ELSE
                $mens_ttir = $CHR(50+cod_fun)+$INTB(er_rob.no.ok)
        END

    ;TRANSMISION DE MENSAJES
        WRITE (lu_est) $mens_ttir, /S   ;se envía el mensaje a través de la unidad lógica
        ;se comprueba si hay error en la transmisión. En ese caso se restablece la conexión
        IF IOSTAT(lu_est) <= 0 THEN
                TYPE "ERROR en la transmisión de "
                TYPE ASC($MID($mens_ttir,1,1))
                DETACH (lu_est)          ;se libera el canal de comunicación
                GOTO 10          ;se restablece la conexión
        END

        hay_syst_err = FALSE  ;indica que no ha habido error

        GOTO 20                   ;espera a una nueva orden
    .END                          ;fin del programa de comunicación
```

El programa de control del robot se encarga de ejecutar los códigos de función pasados por el ordenador en $cod_fun y devuelve el mensaje $mens_ttir. Según el código de función recibido, el robot puede realizar tres tareas : inicialización, tarea de *pick & place* y parada.

En el presente ejemplo sólo se muestra la programación de la tarea de *pick & place*, pudiendo el lector realizar el ejercicio de completar las otras dos tareas. El programa de control, podría tener el siguiente aspecto:

```
.PROGRAM robot()

;INICIALIZACION DE LAS VARIABLES
altura = 20          ;indica la altura de aproximación y salida en milímetros
rapido = 150         ;indica la velocidad rápida
lento = 30           ;indica la velocidad lenta
```

```
;PUNTO DE REENTRADA TRAS SYST_ERR
TYPE "El programa de control ha sido relanzado"

REACTE errores      ;habilita el proceso asíncrono de tratamiento de errores

20  WAIT(hay_cod_fun)    ;espera hasta que esté disponible un código de función

    TYPE "Código recibido", cod_fun  ;indica a través del monitor el código que se ha recibido
    CASE cod_fun OF
        VALUE 1 :      ;INICIALIZACION
                ;aquí se incluiría el código de inicialización
        VALUE 2 :      ;PARADA
                ;aquí se incluiría el código correspondiente a la parada
        VALUE 3 :      PICK & PLACE
        TYPE "Recibida la orden de PICK & PLACE"
        ATTACH (0)   ;Asigna al programa la tarea número 0

        SET pos_in = TRANS(431,610,523,0,180,45);indica el punto de recogida de las piezas
        SET pos_dej = TRANS(227,-548,712,0,180,45);indica el punto de dejada de las piezas
        ;comienza la operación de picking
        SPEED rapida ALWAYS         ;selecciona la velocidad
        ACCEL 90, 75                ;selecciona la aceleración
        APPRO pos_in, altura   ;se aproxima al punto de recogida a una distancia "altura"
        SPEED lenta                 ;reduce la velocidad
        MOVE pos_in                 ;se mueve al punto de recogida
        CLOSEI                      ;cierra la pinza
        DEPARTS altura              ;se separa hasta una distancia "altura"

        ;operación de dejada de la pieza
        SPEED rapida ALWAYS         ;selecciona la velocidad
        ACCEL 90,75                 ;selecciona la aceleración
        APPRO pos_dej, altura  ;se aproxima al punto de dejada a una distancia "altura"
        SPEED lenta                 ;reduce la velocidad
        MOVE pos_dej                ;se mueve al punto de dejada
        OPENI                       ;abre la pinza
        DEPARTS altura              ;se separa hasta una distancia "altura"
    .END  ;fin de la instrucción CASE

hay_cod_fun = FALSE           ;indica que ya no hay código de función
GOTO 20                       ;regresa al estado de espera
.END                          ;fin del programa de control
```

Por último, se muestra un ejemplo de cómo sería el programa de tratamiento de errores.

```
.PROGRAM errores()

    TYPE "RUTINA DE TRATAMIENTO DE ERRORES DEL SISTEMA V+"
    TYPE "Error número : ", ERROR(-1)
    TYPE "Mensaje : ", $ERROR(ERROR(-1))

    error = ABS(ERROR(-1))          ;asigna el error que se ha producido
    $cod_er = $INTB(error) ;comprueba cuál es el error
    hay_syst_err = TRUE             ;indica que se ha producido un error

    hay_cod_fun = FALSE  ;indica que no hay código de función

    EXECUTE 2 err_robot()           ;ejecuta el programa err_robot asignándole la tarea 2
    CYCLE.END 2
    RETURN
.END                                ;fin del programa de tratamiento de errores

.PROGRAM err_robot()
    ABORT 1                         ;aborta la ejecución del programa de control del robot
    CYCLE.END 1
    EXECUTE 1 robot()               ;ejecuta el programa de control del robot
                                    ;asignándole la tarea #1
    ABORT 2
.END                                ;fin del programa err_robot()
```

8.5. BIBLIOGRAFÍA

[ABB-94]ABB Robotics, *RAPID Programmig Manual*, 1994.

[ADEPT-89] Adept Technology Inc., *Adept V+ Programming Manual*, 1989.

[AMBLER-86] Ambler, A.P, Cameron, S.A. y Corner, D.F., "Augmenting the RAPT Robot Language", en: *Languages for Sensor-Based Control in Robotics* (U.Rembold y K. Hörmann Ed.), pp. 305-317. Springer-Verlag, Berlin, 1986.

[BARRIENTOS-86] A. Barrientos, *Sistema operativo y lenguaje de programación para robots dotados de sensores*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid.

[BLUME-86] Blume, C. y Jakob, W., *Programming Languages for Industrial Robots*, Springer-Verlag, Berlin, 1986.

[BONNER-82] Bonner, S. y Shin,K.G, "A Comparative Study of Robot Languages", *Computer,* 12, pp 82-96, 1982

Capítulo 9

CRITERIOS DE IMPLANTACIÓN DE UN ROBOT INDUSTRIAL

Del dicho al hecho hay gran trecho.
Refranero popular

Un robot industrial raramente trabaja como un elemento aislado, sino que forma parte de un proceso de fabricación que incluye muchos otros equipos. El robot, parte principal de la denominada célula de trabajo robotizada, debe en general interactuar con otras máquinas, formando parte de un estructura de fabricación superior. En este capítulo se abordarán, tanto desde un aspecto técnico como económico, aquellos temas relacionados con la implantación de un robot en un entorno industrial.

El primer aspecto considerado es el diseño de la célula desde el punto de vista del *lay-out* (esquema de disposición de equipos, máquinas y demás elementos de la planta) y de la arquitectura de control. Se trata de una cuestión importante para poder extraer las máximas prestaciones a un robot industrial durante su explotación, como también lo es el escoger el robot adecuado para una determinada aplicación. En este capítulo se dan algunas indicaciones y se explican aquellas cuestiones a considerar para tomar dicha decisión. La seguridad de la célula robotizada también es tratada de forma especial, pues obliga a tener en cuenta consideraciones importantes en la fase de diseño, desarrollo y explotación. Se exponen también algunas indicaciones de cómo realizar la justificación económica de la implantación de una célula robotizada. Finalmente, se muestra una imagen actualizada de cuál es la situación del mercado de robots en España.

[CRAIG-89] Craig, JJ., "Robot Programming Languages and Systems", en: *Introduction to Robotics Mechanics and Control*, 2ª edicion. Addison-Wesley Publishing Company, Reading, Massachusets, 1989.

[FELIÚ-85] Feliú, V., Barrientos, A. y Puente, E.A., "A Low Level Language for Assembly Robots", *International Symposium on Robot Control (SYROCO'85)*, pp.37-44, 1985.

[FINKEL-74] Finkel, R. et al., "AL, A Programming System for Automation", *Memo AIM-243*, Stanford Artificial Intelligence Laboratory, 1974.

[FU-88] Fu, K.S., Gonzalez, R.C. y Lee, C.S.G., "Lenguajes de Programación de Robots", en: *Control, Detección, Visión e Inteligencia*, MacGraw-Hill/Interamericana, Madrid, 1988.

[KOUTSOU-81] Koutsou,A., "A Survey of Model-Based Robot Programming Languages", *DAI Working Paper No. 108*. Dept of Artificial Intelligence, University of Edimburgh, 1981.

[LIEBERMAN-77] Lieberman, L.I. y Wesley, M.A., "AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly", *IBM Journal of Research & Development*, 4, pp. 321-333, 1977.

[LOZÁNO-82] Lozano-Pérez, T., "Robot Programming", *Memo AIM*-698, Massachusetts Institute of Technology, 1982.

[LÓZANO-76] Lózano-Pérez, T., "The Design of a Mechanical Assembly System", *AI Lab. Technical Report*, Massachusetts Institute of Technology, 1976.

[POPPLESTONE-78] Popplestone, R.J., Ambler, A.P y Bellos, I., "RAPT: A Language for Describing Assemblies", *The Industrial Robot*, 5, pp. 131-137, 1978.

[REMBOLD-86] Rembold, U., "Programming of Industrial Robots, Today and in the Future", en: *Languages for Sensor-Based Control in Robotics*,(U.Rembold y K. Hörmann Ed.), pp.3-24. Springer-Verlag, Berlin, 1986.

[REMBOLD-85] Rembold, U., Blume, C. y Frommherz, B.J., "The Proposed Robot Software Interfaces SRL and IRDATA", *Robotics & Computer-Integrated Manufacturing*, Vol.2 No 3/4, pp. 219-225, 1985.

[RODRIGUEZ-94] Rodríguez, C.F.., *Planificación y control de sistemas multirobot en tareas coordinadas*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1994.

[STAÜBLI-92] Staübli, S.A., *Guía VAL II*, 1992.

[TAYLOR-82] Taylor, R.H., Summers, P.D. y Meyer, J.M., "AML: A Manufacturing Language", *The International Journal of Robotics Research*, Vol.1 No 3, pp. 19-56, 1982.

[VOLZ-88] Volz, R.A., "Report of the Robot Programming Language Working Group: NATO Workshop on Robot Programming Languages", *IEEE Journal of Robotics and Automation*, Vol.2 No 1, pp. 86-90, 1988.

9.1. DISEÑO Y CONTROL DE UNA CÉLULA ROBOTIZADA

El proyecto e implantación de un sistema robotizado implica la consideración de un gran número de factores, que van desde el posible rediseño del producto, hasta la definición detallada del *lay-out* o plano de implantación del sistema (Figura 9.1).

Junto con la selección más adecuada del robot para la aplicación, habrá que definir, e incluso diseñar, los elementos periféricos pasivos (mesas, alimentadores, utillajes, etc.) o activos (manipuladores secuenciales, máquinas CN, etc.) que intervienen en la célula, y situarlos físicamente en el sistema.

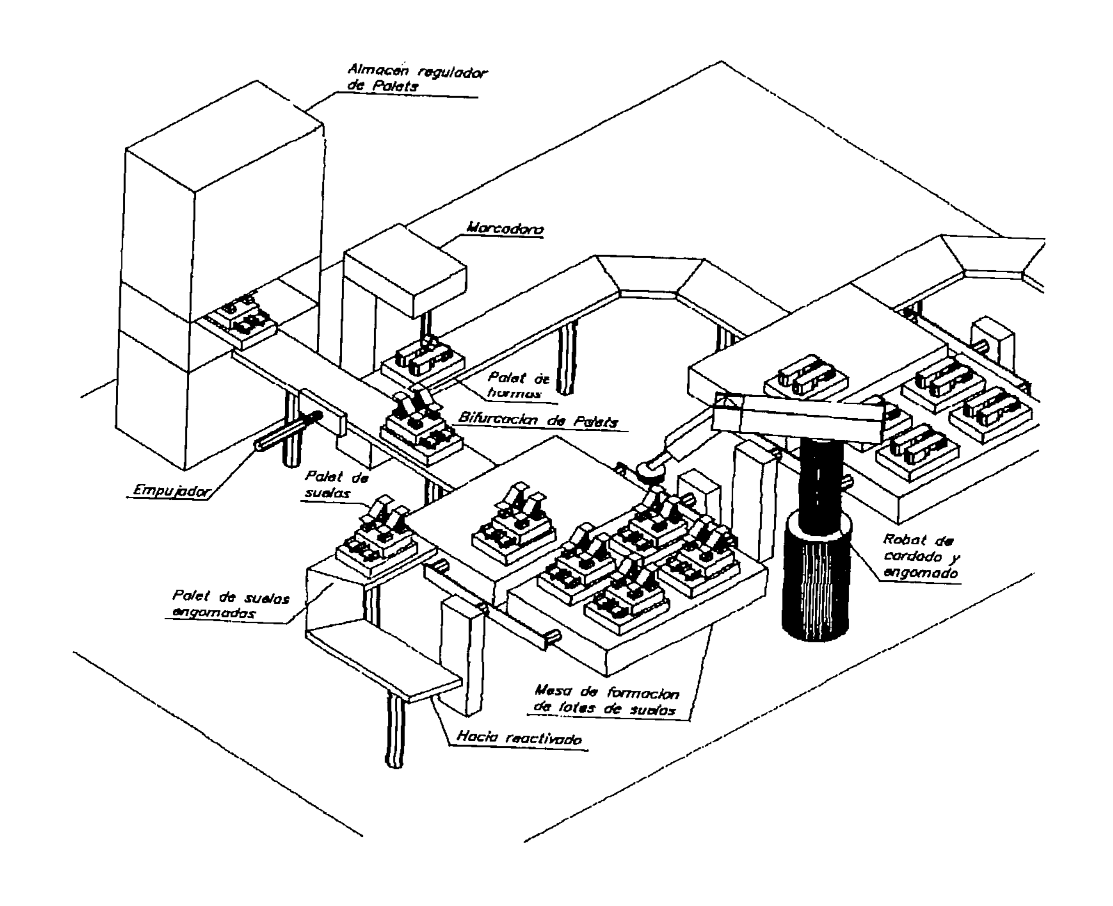


Figura 9.1. Ejemplo del *lay-out* de una célula robotizada de fabricación de calzado.

Asimismo será preciso definir y seleccionar la arquitectura de control, tanto hardware como software, que todo sistema flexible de fabricación debe incluir.

La definición del *lay-out* del sistema es un proceso iterativo del que debe resultar la especificación del tipo y número de robots a utilizar, así como de los elementos periféricos, indicando la disposición relativa de los mismos. En este proceso iterativo es clave la experiencia del equipo técnico responsable del diseño. La utilización de herramientas informáticas, como sistemas CAD, simuladores específicos para robots y simuladores de sistemas de fabricación flexible facilitan enormemente esta tarea.

Un simulador de sistemas robotizados permite de manera rápida y eficiente evaluar las diferentes alternativas en cuanto al robot a utilizar y la disposición física de todo el sistema. Utilizando la interacción gráfica, se puede analizar qué robot de los existentes en la librería del programa se adapta mejor a la tarea programada, detectando posibles colisiones y verificando el alcance [RODRIGUEZ-93]. Un simulador de sistemas de fabricación flexible permite dimensionar adecuadamente la célula, informando sobre su productividad, rendimiento y comportamiento ante cambios de la demanda o situaciones imprevistas (averías, cambios en el producto, etc.). Asimismo permite ensayar diferentes estrategias de control de la célula encaminadas a optimizar su funcionamiento [CACI-95].

9.1.1. Disposición del robot en la célula de trabajo

A la hora de decidir la disposición del robot en la célula, cabe plantearse cuatro situaciones básicas:

Robot en el centro de la célula

En esta disposición el robot se sitúa de modo que quede rodeado por el resto de elementos que intervienen en la célula. Se trata de una disposición típica para robots de estructura articular, polar, cilíndrica o SCARA, en la que se puede aprovechar al máximo su campo de acción, que presenta una forma básica de esfera.

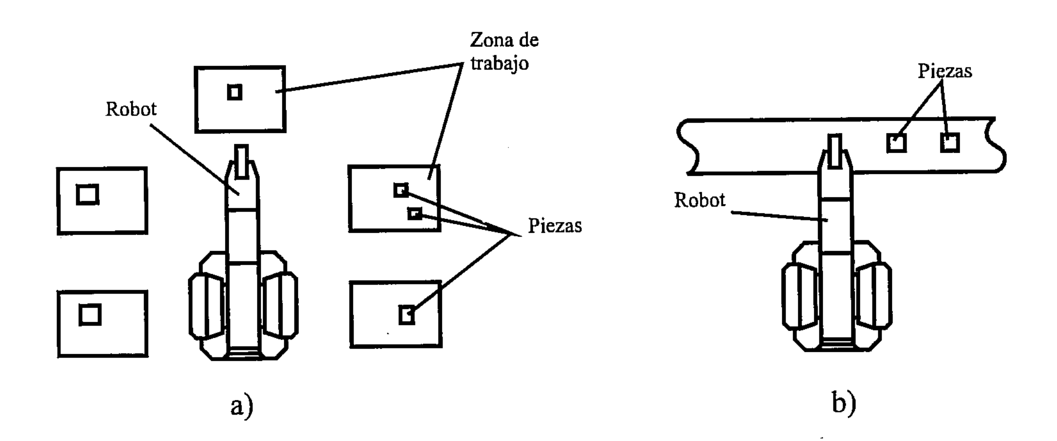


Figura 9.2. Disposición del robot en a) el centro de la célula y b) en línea.

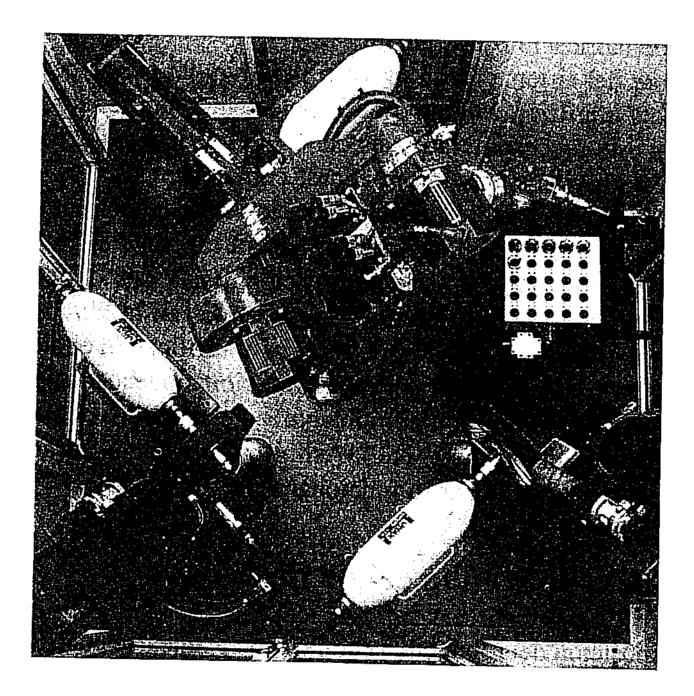


Figura 9.3. Robot en el centro de una célula de pulido del piezas de reloj. (Cortesía de KUKA.)

La disposición del robot en el centro se usa frecuentemente en aquellas aplicaciones en las que un robot sirve a una o varias máquinas (carga-descarga de máquinas herramientas), así como en las aplicaciones de soldadura al arco, paletización o ensamblado, en las que el robot debe alcanzar diversos puntos fijos dentro de su área de trabajo (Figura 9.2-a y Figura 9.3).

Robot en línea

Cuando uno o varios robots deben trabajar sobre elementos que llegan en un sistema de transporte, la disposición de robots en línea es la más adecuada. (Figura 9.2-b.)

El ejemplo más representativo de esta disposición son las líneas de soldadura de carrocerías de vehículos, en las que éstos pasan secuencialmente frente a sucesivos robots alineados, cada uno de los cuales realiza una serie de puntos de soldadura.

En este tipo de disposición cabe diferenciar que el transporte sea de tipo intermitente o continuo. En el primer caso, en un momento determinado cada robot tiene delante una pieza sobre la que realiza las operaciones establecidas. Una vez acabadas éstas, bien se espera a que todos los robots finalicen sus tareas, para que entonces el sistema de transporte avance un puesto, o bien, si el sistema lo permite, da salida a la pieza que proceda, quedando disponible para recibir una nueva.

Si el transporte es continuo, esto es, si las piezas no se detienen delante del robot, éste deberá trabajar sobre la pieza en movimiento, para lo que el transporte deberá limitar su velocidad de modo que la pieza quede dentro del alcance del robot durante al menos el tiempo de ciclo.

Robot móvil

En ocasiones es útil disponer al robot sobre una vía que permita su desplazamiento lineal de manera controlada (Figura 9.4). Esta posibilidad permite, por ejemplo, seguir el movimiento de la pieza en el caso de que ésta se desplace sobre un sistema de transporte continuo, de modo que la posición relativa entre pieza y robot durante el tiempo que dura el proceso se mantenga fija. Para ello es necesario una precisa sincronización de los dos movimientos: transporte de la pieza y transporte del robot, lo que se puede conseguir con un acoplamiento mecánico desembragable o mediante un adecuado control en cadena cerrada de ambos transportes. En cualquier caso, una vez acabado el procesamiento de la pieza, el robot debe regresar rápidamente a su posición inicial para recibir una nueva.

Otra situación en la que el empleo del robot con capacidad de desplazamiento lineal es particularmente ventajosa es cuando éste debe cubrir un elevado campo de acción. Por ejemplo, en la pintura de carrocerías de coches, el dotar al robot de este grado de libertad adicional permite que dos robots de dimensiones medias (2 metros de radio de alcance aproximadamente) lleguen con la orientación adecuada a todos los puntos de proyección correspondientes a un coche.

Asimismo, esta disposición del robot puede utilizarse cuando éste tenga que dar servicio a varias máquinas (por ejemplo, para carga-descarga de máquinas herramientas), obteniendo el máximo rendimiento del robot.

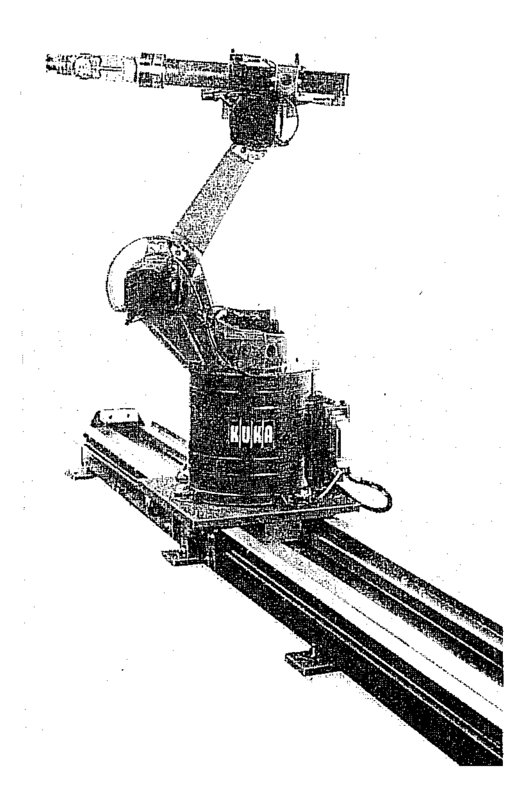


Figura 9.4. Disposición de robot móvil sobre rail. (Cortesía de KUKA.)

Robot suspendido

La disposición del robot suspendido es la intrínseca de un robot tipo pórtico en la que éste queda situado sobre el área de trabajo. Pero además de esta estructura de robot, es posible colocar un robot articular invertido sobre la célula (Figura 9.5).

Las ventajas fundamentales que se obtienen en este segundo caso son las de un mejor aprovechamiento del área de trabajo, pues de este modo el robot puede acceder a puntos situados sobre su propio eje vertical.

Las operaciones típicas donde se utiliza el robot suspendido son en aplicación de adhesivos o sellantes, proyección de material (pintura, acabado superficial, etc.), corte (chorro de agua, láser, etc.) y soldadura al arco. Algunos fabricantes incluso han desarrollado modificaciones sobre sus robots básicos, tanto en su estructura mecánica como en su control, para facilitar el uso del robot en esta disposición.

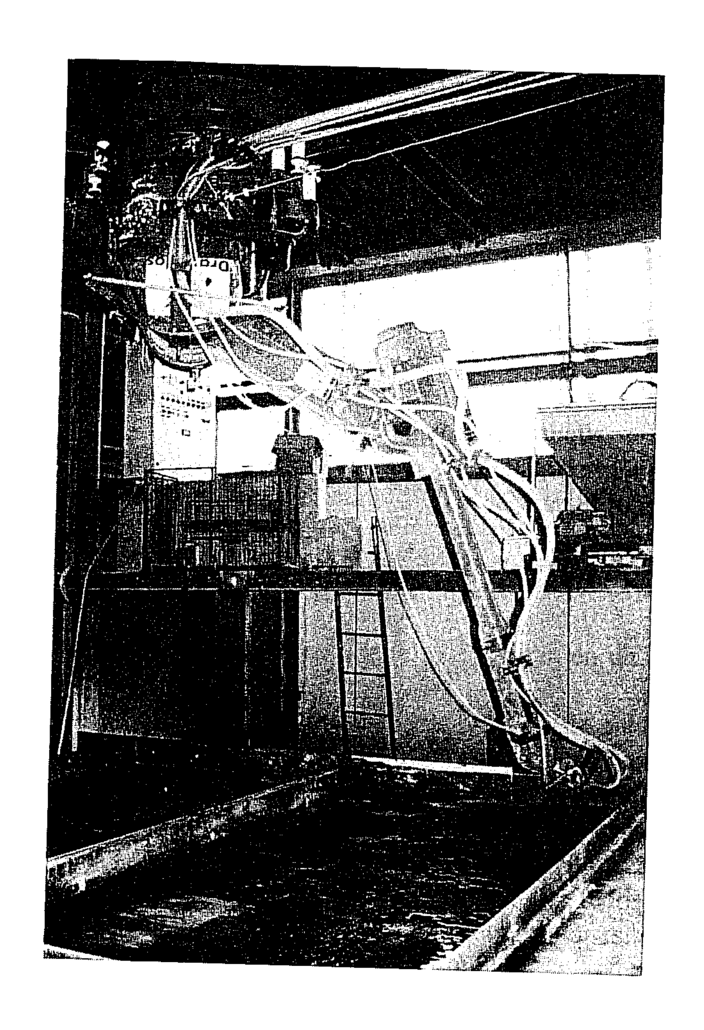


Figura 9.5. Robot IRB 3200 proyectando cemento en disposición suspendido. (Cortesía de DISAM-UPM.)

9.1.2. Características del sistema de control de la célula de trabajo

Una vez establecidos los elementos operativos de la célula y su disposición en la planta, el segundo factor crítico en la definición de la célula robotizada es la especificación del sistema de control.

Una célula robotizada debe responder a las premisas de flexibilidad y automatización que justifican su empleo. Para ello es imprescindible el establecimiento de un buen sistema de control que deberá realizar, entre otras, las siguientes funciones.

- **Control individual** de cada una de las máquinas, transportes y demás dispositivos, incluidos robots, que compongan la célula.
- **Sincronización** del funcionamiento de los diferentes dispositivos entre sí.
- **Detección, tratamiento y recuperación** si es posible de las situaciones anómalas de funcionamiento que puedan presentarse.
- **Optimización del funcionamiento** conjunto de los dispositivos de la célula, distribuyendo si es posible las funciones de manera dinámica, para así evitar paradas por espera o acciones innecesarias.
- **Interfaz con el usuario**, mostrando la información adecuada para que en todo momento se conozca con el detalle necesario el estado del sistema, así como permitiendo que el operador acceda, con las restricciones pertinentes, al funcionamiento del mismo.
- **Interfaz con otras células**, para permitir la sincronización entre ellas, optimizando el funcionamiento de un sistema de fabricación flexible compuesto por varias células.
- **Interfaz con un sistema de control superior** que realiza básicamente funciones de supervisión y actualización de programas cuando se diese un cambio en la producción.

Estas funciones, que pueden ser necesarias en mayor o menor medida en el control de una célula robotizada, se implementarán en un hardware que será preciso definir y dimensionar. En aquellas ocasiones en las que la simplicidad de la célula lo permite, el propio controlador del robot podrá simultanear las funciones propias de control de sus ejes con el mando del resto de los dispositivos. Para ello utilizará sus entradas/salidas digitales/analógicas, o si fuera necesario la posibilidad de incorporar ejes externos servocontrolados.

Si por el contrario la célula incluye un mayor número de dispositivos, algunos de los cuales incluso disponen de su propio controlador, será preciso disponer de una estructura jerarquizada en la que un elemento central (ordenador, PLC, etc.) mantenga una comunicación con el resto de los controladores.

En cualquier caso, el correcto dimensionamiento del control de la célula es tarea que no debe desestimarse, pues repercutirá tanto en el coste de implantación y desarrollo de la misma como en el de explotación.

9.2. CARACTERÍSTICAS A CONSIDERAR EN LA SELECCIÓN DE UN ROBOT

Cuando se desea robotizar un determinado proceso, el equipo de técnicos responsable de esta tarea debe seleccionar el robot más adecuado. Para ello recurrirá a su experiencia y buen criterio, escogiendo, dentro del amplio mercado de robots existente, aquel que mejor responda a las características necesarias y buscando siempre el adecuado compromiso entre precio y prestaciones.

Tabla 9.1. Características a tener en cuenta para la selección de un robot

Caract. geométricas	· Área de trabajo · Grados de libertad · Errores de posicionamiento · Distancia tras emergencia · Repetividad · Resolución · Errores en el seguimiento de trayectorias · Calidad de una línea recta, arco,... · Precisión cuando se mueve el mínimo incremento posible
Caract. cinemáticas	· Velocidad nominal máxima · Aceleración y deceleración
Caract. dinámicas	· Fuerza · De agarre · Carga máxima · Control de fuerza-par · Frecuencia de resonancia
Tipo movimientos	· Movimientos punto a punto · Movimientos coordinados · Trayectorias continuas (CP)
Modo programación	· Enseñanza (guiado) · Textual
Tipo accionamiento	· Eléctrico (c. alterna o c. continua) · Neumático · Hidráulico
Comunicaciones	· E/S Digitales/Analógicas · Comunicaciones línea serie
Servicio proveedor	· Mantenimiento, Servicio Técnico, Cursos de formación
Coste	

La selección del robot más idóneo debe hacerse valorando una gran variedad de características, siendo éste un proceso de difícil sistematización. Sin embargo, en general puede ser suficiente con considerar un conjunto limitado.

En los catálogos de robots, los fabricantes proporcionan los valores de las prestaciones de sus productos. Estos valores están con frecuencia medidos en condiciones óptimas, diferentes de las reales; no obstante, sirven como medida comparativa para la selección del robot. Por otra parte, aunque hasta el momento no existe una homologación oficial para los robots industriales que garanticen la veracidad de los datos de catálogo, sí existen laboratorios con acreditado renombre en los que se realizan una serie de test orientados a cuantificar las características de los robots con elevada exactitud. Este es el caso del instituto alemán IPA (Stuttgart).

Se van a comentar a continuación las características más destacadas que deben ser consideradas a la hora de seleccionar un robot para una determinada aplicación, y que han sido recogidas de manera resumida en la Tabla 9.1.

9.2.1. Área de trabajo

El área de trabajo o campo de acción es el volumen espacial al que puede llegar el extremo del robot. Este volumen está determinado por el tamaño, forma y tipo de los eslabones que integran el robot (Figura 9.6), así como por las limitaciones de movimiento impuestas por el sistema de control. Nunca deberá utilizarse el efector colocado en la muñeca para la obtención del espacio de trabajo, ya que se trata de un elemento añadido al robot, y en el caso de variar el efector el área de trabajo se tendría que calcular de nuevo.

En los catálogos suministrados por los fabricantes se suele indicar el área de trabajo mediante un dibujo acotado. Cuando la información es de tipo numérico, el área de trabajo se indica mediante el rango de recorrido de cada articulación.

El robot debe elegirse de modo que su área de trabajo (o campo de acción) le permita llegar a todos los puntos necesarios para llevar a cabo su tarea. En este sentido, no debe olvidarse la necesidad de incluir entre los puntos a acceder los correspondientes a puntos de recogida de piezas (alimentadores), mesa de trabajo, puntos de salida de piezas, etc.

El que el robot pueda acceder a todo el espacio de trabajo no significa que lo pueda hacer con cualquier orientación. Existirán un conjunto de puntos, los más alejados y los más cercanos, que únicamente se podrán acceder con unas orientaciones determinadas, mientras que otros puntos admitirán cualquier orientación.

Se ha de tener en cuenta también la posible existencia de los denominados puntos singulares, que ya fueron explicados en el epígrafe 4.3.3. Se trata de puntos con una determinada orientación en el espacio sobre los que, por ejemplo, no es posible realizar una trayectoria rectilínea, bien sea porque su ejecución implicaría el movimiento a velocidad infinita de uno de los ejes, bien porque el valor de los ejes en ese punto con esa orientación se encuentre indeterminado.

La disposición óptima de todos los elementos que compondrán la célula junto con el robot, es una delicada tarea por el gran número de variables a considerar. No basta con asegurarse de que todos los puntos necesarios quedan dentro del campo de acción, sino que se deberá verificar que una vez situados los demás componentes de la célula, el robot no colisione con ellos al efectuar sus movimientos.

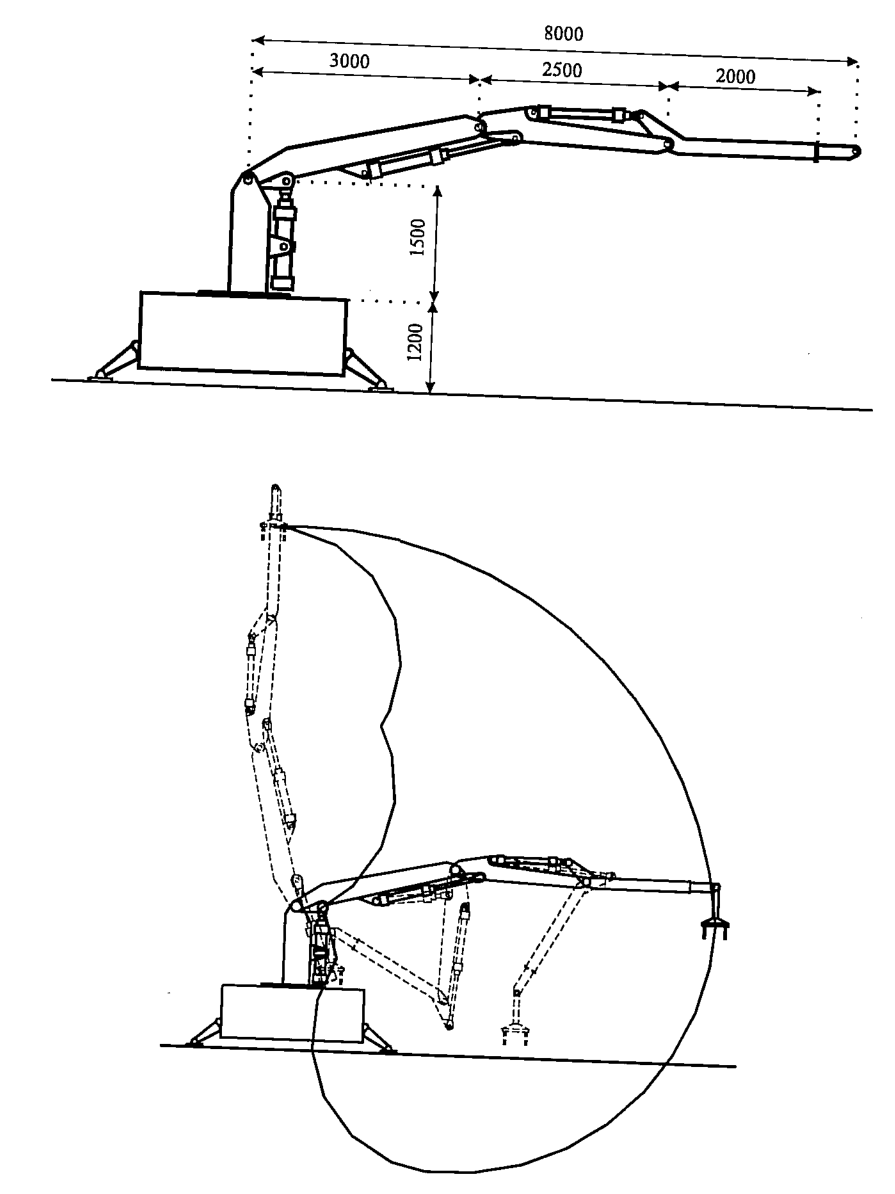


Figura 9.6. Configuración y sección del área de trabajo del robot ROCCO.

Por este motivo, es de gran ayuda el empleo de programas de simulación gráfica (ROBCAD, GRASP, TOROS, etc.), que dotados de un sistema de diálogo interactivo con el usuario, permiten seleccionar mediante ensayo y error la disposición óptima de la célula, pudiéndose incluso ensayar diferentes robots almacenados para tal efecto en memoria (Figura 9.7).

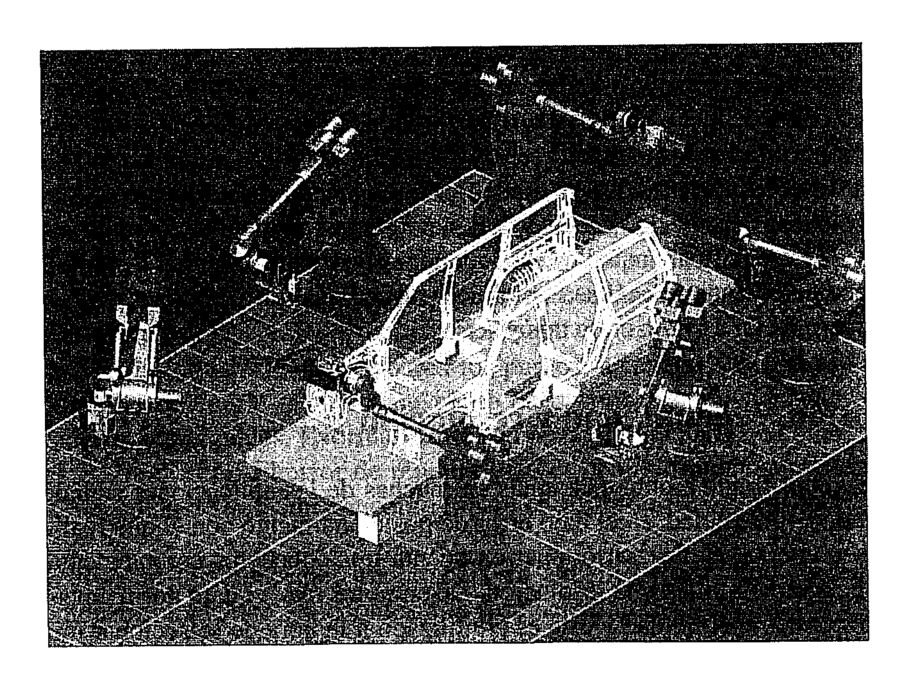


Figura 9.7. Célula de soldadura en la industria de automoción simulada con ROBCAD. (Cortesía de TECNOMATIX.)

9.2.2. Grados de libertad

El número de grados de libertad con que cuenta un robot (GDL) determina la accesibilidad de éste y su capacidad para orientar su herramienta terminal. Es relativamente frecuente que el número de GDL de los robots comerciales coincida con el número de articulaciones, es decir, que cada articulación representa un GDL.

La elección del número de grados de libertad necesarios viene determinada por el tipo de aplicación. Así, en muchas operaciones de manipulación, (*pick & place*, paletizado) los objetos se recogen y depositan sobre planos horizontales. En estos casos, un robot con 3 GDL para posicionar, y a lo sumo uno más para orientar (giro en torno a un eje vertical), es suficiente. Sin embargo, en otras aplicaciones, es preciso orientar la herramienta en el espacio o acceder a posiciones complicadas, siendo precisos 6 o incluso más grados de libertad. Aplicaciones típicas que precisan de 6 GDL pueden ser la pintura, la soldadura al arco o la aplicación de sellantes.

Con cierta frecuencia los fabricantes de robots proporcionan un número determinado de grados de libertad ampliables (en 1 o 2) de manera opcional. Este grado extra se añade al robot en unos casos en su extremo y en otros en su base.

En general hay que considerar que el aumento del número de grados de libertad lleva parejo un considerable aumento del coste del robot. En ocasiones, éste puede suplirse con el empleo de un utillaje apropiado y con el rediseño de las piezas que entran a formar parte de la instalación. Así, en tareas en las

que es preciso acceder a piezas grandes o complejas, éstas pueden ir montadas sobre mesas posicionadoras u orientadoras, que disponen de un número finito o infinito de posiciones alternativas, controladas de manera sincronizada con el robot. En el primer caso, el accionamiento puede ser neumático o hidráulico, siendo su coste comparativamente reducido. En el segundo caso (posicionamiento continuo), el precio de estos periféricos auxiliares es elevado, debiéndose escoger con cuidado la solución a adoptar.

En cualquier caso, y para tareas muy específicas, se pueden desarrollar robots con más de 6 GDL como configuración básica. Se les suele denominar robots redundantes, aunque esta terminología se aplica de forma más específica a aquellos robots que poseen más GDL que los que implica la tarea que realizan, independientemente de cuál sea el número de los mismos. En la actualidad los valores típicos en cuanto al número de GDL varía entre tres y seis.

9.2.3. Precisión, repetibilidad y resolución

Las ventajas del robot frente a otras máquinas en muchas de las aplicaciones actuales se basan además de en la flexibilidad y velocidad, en el bajo error de posicionamiento con el que realizan su trabajo. Para la definición de este error es necesario tener en cuenta tres conceptos complementarios entre sí, como son: la precisión, la repetibilidad y la resolución. De entre los tres, el dato normalmente suministrado por los fabricantes es el de repetibilidad y éste es el utilizado a la hora de seleccionar un robot u otro por su exactitud.

La Figura 9.8 presenta gráficamente estos tres conceptos. El origen de cada uno de ellos es diferente; su definición y causa se indica a continuación:

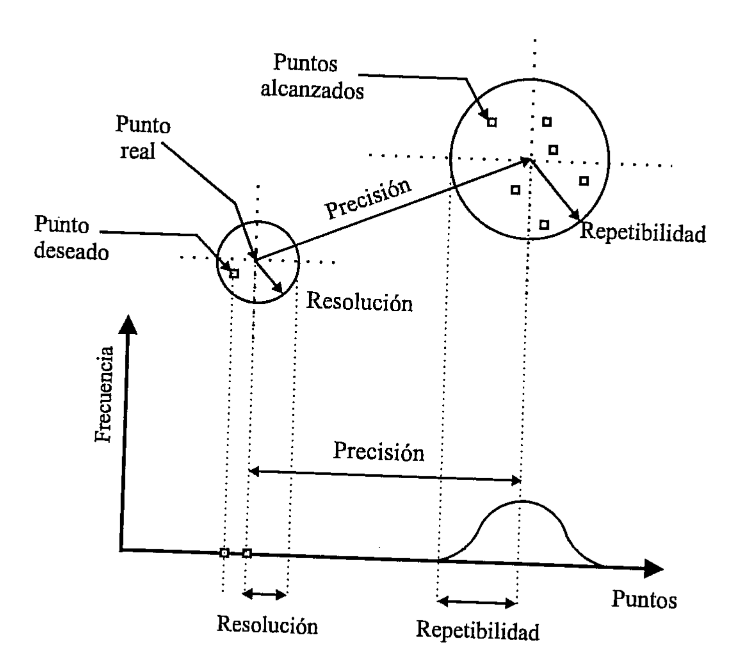


Figura 9.8. Visualización de los conceptos de resolución, precisión y repetibilidad.

- **Resolución:** Mínimo incremento que puede aceptar la unidad de control del robot. Su valor está limitado por la resolución de los captadores de posición y convertidores A/D y D/A, por el número de bits con los que se realizan las operaciones aritméticas en la CPU. y por los elementos motrices, si estos son discretos (motores paso a paso, sistemas neumáticos todo nada, etc.)
- **Precisión**: Distancia entre el punto programado (normalmente de manera textual) y el valor medio de los puntos realmente alcanzados al repetir el movimiento varias veces con carga y temperatura nominales. Su origen se debe a errores en la calibración del robot (punto de sincronismo por ejemplo), deformaciones por origen térmico y dinámico, errores de redondeo en el cálculo de la transformación cinemática (especialmente en las cercanías de puntos singulares), errores entre las dimensiones reales y teóricas del robot, etc.
- **Repetibilidad**: Radio de la esfera que abarca los puntos alcanzados por el robot tras suficientes movimientos, al ordenarle ir al mismo punto de destino programado, con condiciones de carga, temperatura, etc., iguales. (Normalmente se considera la banda que abarca el 99% de los puntos respecto a la media.) El error de repetibilidad es debido fundamentalmente a problemas en el sistema mecánico de transmisión como rozamientos, histéresis, zonas muertas (*backlash*).

El error de repetibilidad tiene especial importancia en aquellos robots que son programados por aprendizaje, pues entonces no afectan los debidos a la resolución ni precisión. Los valores normales de error de repetibilidad de robots industriales comerciales varían entre los ±2 milímetros y ±0,01 milímetros.

En el valor total del error de posicionamiento de un robot, afectan una serie de factores, como la longitud de sus brazos, carga manejada, tipo de estructura (la cartesiana no precisa transformación homogénea, evitándose errores de cálculo), que pueden dar una idea general sobre la calidad del posicionamiento final de su extremo. Así, por lo general, los robots cartesianos y los de reducidas dimensiones son más precisos (en el sentido global) que otros como articulares o robots de gran envergadura.

Otras medidas relativas a los posibles errores de posición de un robot son las relacionadas con la precisión con que un robot, que disponga de capacidad para ello, recorre una determinada trayectoria programada, por ejemplo una línea recta. En este caso, los posibles errores se ven afectados por las mismas causas que los anteriores más por aquellas derivadas del algoritmo interpolador de la trayectoria y de su control dinámico.

Así, el número de puntos con que se interpole una trayectoria determinará la precisión con que el robot la sigue. Este número de puntos está limitado por el tiempo de cálculo de la transformación inversa (dependiente de la estructura del robot, potencia de cálculo del sistema de control, etc.), así como por la velocidad con que se desee recorrer la trayectoria programada (lógicamente una menor velocidad posibilitará un mayor número de puntos interpolados).

9.2.4. Velocidad

Como ya se ha indicado, la velocidad a la que puede moverse un robot y la carga que transporta, están inversamente relacionados. Tanto es así que en muchas ocasiones los datos proporcionados en catálogo sobre la velocidad de movimiento del robot se dan para diferentes valores de la carga a transportar. De

igual forma y como es lógico, también suele existir una relación de orden inverso entre el error de posicionamiento y la velocidad del robot.

La velocidad de movimiento de un robot puede darse por la velocidad de cada una de sus articulaciones o por la velocidad media de su extremo, siendo esta última más útil para el usuario, pero más imprecisa. El valor de la velocidad nominal de movimiento de un robot es un dato relevante para el cálculo de los tiempos de ciclo, sobre todo en robots destinados a tareas de manipulación o ensamblaje. No obstante, hay que considerar que el dato proporcionado normalmente corresponde a la velocidad nominal en régimen permanente. Para alcanzar este régimen es preciso que el movimiento del robot sea suficientemente largo. En otro caso, los tiempos de arranque y parada son proporcionalmente más significativos que el correspondiente al movimiento a velocidad nominal.

En la práctica, en la mayoría de los casos los movimientos del robot son rápidos y cortos, con lo que la velocidad nominal es alcanzada en contadas ocasiones. Por este motivo, la medida del tiempo de ciclo no puede ser obtendia a partir de la velocidad, siendo ésta una valoración cualitativa del mismo. En vez de este dato, algunos robots indican el tiempo empleado en realizar un movimiento típico (un *pick & place*, por ejemplo).

Los valores habituales de velocidad del extremo oscilan entre 1 y 4 m/s con carga máxima.

9.2.5. Capacidad de carga

La capacidad de carga del robot a seleccionar para una determinada tarea viene condicionada por el tamaño, la configuración y el sistema de accionamiento del propio robot. Por otra parte, al evaluar la carga a manipular por el robot debe considerarse el peso de las piezas a manipular y el propio peso de la herramienta o pinza que emplee el robot colocada sobre la muñeca (en muchas ocasiones superior al de los propios objetos). Se debe tener en cuenta además de la carga, el momento que la pieza a transportar genera en el extremo del robot. Para ello el fabricante puede proporcionar un cuadro en el que se indica la disminución de la posible carga a transportar para no disminuir prestaciones a medida que el centro de gravedad de la misma se aleja del centro de la muñeca.

El dato que normalmente se proporciona en la hoja de características del robot, corresponde a la carga nominal que éste puede transportar sin que por ello disminuyan sus prestaciones dinámicas, y siempre considerando la configuración del robot más desfavorable. Sin embargo, es posible aumentar esta carga hasta un cierto límite, siempre y cuando se pueda admitir una disminución en la velocidad de los movimientos del robot e incluso en su precisión.

Los valores más frecuentes de capacidades de carga varían entre 5-50kg, aunque se pueden encontrar robots que transporten más de media tonelada.

9.2.6. Sistema de control

La potencia de la unidad de control del robot determina en gran medida sus posibilidades. Las características del control del robot hacen referencia por una parte a sus posibilidades cinemáticas (tipo de trayectorias) y dinámicas (prestaciones dinámicas del robot), y por otra parte a su modo de programación.

En cuanto a las posibilidades cinemáticas es muy importante tener en cuenta la aplicación a realizar. Para muchas aplicaciones (*pick & place*, por ejemplo), es suficiente con un control del movimiento

punto a punto (PTP) en el que sólo es relevante el punto final a alcanzar por el robot y no el camino seguido. En otras, por el contrario, la trayectoria continua (CP) descrita por el extremo del robot es fundamental (soldadura al arco). Casi todos los robots incorporan la posibilidad de realizar trayectorias en línea recta y con interpolación circular. Estas posibilidades vienen normalmente indicadas en las especificaciones técnicas del robot.

Un primer dato relativo al control dinámico de un robot, es el que indica si éste se efectúa en cadena abierta o cerrada. El primer caso no es frecuente si bien se emplea cuando no se prevén grandes inercias. Su implementación se realiza normalmente con motores paso a paso, simplificando notablemente la complejidad de los algoritmos de control.

Las características del control dinámico del robot, como velocidad de respuesta y estabilidad, son de particular importancia cuando éste debe manejar grandes pesos con movimientos rápidos. En estos casos, un buen control dinámico asegura que el extremo del robot no presente oscilaciones ni errores de posicionamiento. El sobrepasar el punto de destino (*overshoot*) por una elevada inercia, puede originar colisiones de graves consecuencias.

Normalmente las prestaciones del control dinámico no son indicadas explícitamente como una característica a conocer por un posible usuario. Algunos sistemas de control de robots permiten variar, incluso en mitad de la ejecución de un programa, algunas de las características del control dinámico (su acción P o I en caso de que se trate de un servo).

Otra importante característica relacionada con el control dinámico hace referencia a la posibilidad de realizar un control de esfuerzos de manera selectiva en alguna de las articulaciones o ejes cartesianos. Esta posibilidad, que implica el empleo de sensores de esfuerzos, es fundamental en aquellas aplicaciones en las que la pieza manipulada deba entrar en contacto con algún objeto durante la realización de la tarea (ensamblaje, desbarbado, pulido, etc.).

En cuanto a las características relacionadas con el método de programación y las posibilidades que éste ofrece, puede decirse que una primera división entre programación por guiado y programación textual es suficiente como para decidirse sobre el empleo de un robot u otro para una determinada

Tabla 9.2. Carácterísticas deseables en un robot según su aplicación

PINTURA	PALETIZACIÓN
Programación por guiado Campo de acción similar al humano Estructura antropomórfica 6 grados de libertad	Elevada capacidad de carga Relación grande entre área de trabajo y tamaño del robot Control PTP
PROCESADO	ENSAMBLADO
Sistema de programación 5-6 grados de libertad Campo de acción similar al humano Control de trayectoria continua	Elevada precisión y rapidez Campo de acción similar al humano Potencia del sistema de programación Sistema sensorial

aplicación. Así, una aplicación de pintura debe realizarse con un robot cuya programación se efectúe fundamentalmente mediante guiado, y más específicamente, con un sistema en el que la unidad de control memorice automáticamente el camino por el que se desplaza el extremo del robot durante la etapa de programación. Por el contrario, en un robot destinado a tareas de paletizado será aconsejable la programación textual.

Otras características importantes relacionadas con el modo de programación son las relativas al manejo de entradas salidas, posible estructuración de los programas (manejo de subrutinas, bucles, etc.), posibilidad de atención de interrupciones y modificación de trayectorias atendiendo a señales externas, sistemas de programación y control desde un dispositivo externo (computador), etc. Estos aspectos ya se comentan con más detalle en el epígrafe 8.2, dedicado a los sistemas de programación para robots industriales.

Tabla 9.3. Características técnicas de algunos robots comerciales

Fabricante	Modelo	Aplicación	Configuración	GDL	Alcance horizon. (mm)	Carga máxima (kg)	Veloc. máxima (mm / s)	Repetibilidad (mm)	Accionamiento	Control trayec.
ABB	IRB 2400	Uso general	Angular	6	1542	10	4000	0.08	Eléctrico (c.a.)	PTP CP
	TR 5002	Pintura	Angular	6	2574	5	2000	1.0	Eléctrico (c.a.)	PTP CP
Adept	Adept Three	Uso general	SCARA	4	1070	25	11000	0.025	Directo (c.c.)	PTP CP
DEA	BRAVO 2205	Medición	Cartes.	3	x=2525 y=1185	-	500	0.3	Eléctrico (c.c.)	PTP CP
FANUC	ARC Mate 120 / S-12	Sold. Arco	Angular	6	1605	12	135 (°/s)	0.1	Eléctrico (c.a.)	PTP CP
	L-1000	Corte láser	Polar	5	1200	5	-	0.05	Eléctrico (c.a.)	PTP CP
Kremlin	AKR 3000	Pintura	Angular	7	2500	15	2000	1.0	Hidráulico	CP
KUKA	KR 125	Uso general	Angular	6	2410	125	1000	0.2	Eléctrico (c.a.)	PTP CP
Staübli	PUMA560	Uso general Investigar	Angular	6	900	4	4700	0.1	Eléctrico (c.c.)	PTP CP
	RX 90	Uso general	Angular	6	900	12	9800	0.02	Eléctrico (c.c)	PTP CP
Yaskawa	Motoman SK 16-6	Uso general	Angular	6	1775	6	140 (°/s)	0.1	Eléctrico (c.a.)	PTP CP
	Motoman S604	Montaje Manipular	SCARA	4	1950	60	-	0.2	Eléctrico (c.c.)	PTP CP

Existen otras consideraciones, además de las meramente técnicas, a la hora de seleccionar el robot más adecuado para robotizar una proceso. Así, es importante considerar el servicio técnico que proporciona el fabricante (de postventa, mantenimiento, formación, actualización) y que lógicamente variará según la distribución geográfica del lugar de implantación y de la fábrica o delegación del fabricante del robot. También será importante considerar el costo y posibilidad de amortizacíon del robot, pues un robot más barato, y con menos prestaciones, puede resolver correctamente la aplicación en cuestión, pero ser difícilmente adaptable a otras aplicaciones futuras.

La Tabla 9.2 muestra una primera aproximación de las características a tener en cuenta para la elección de un robot en una determinada aplicación. Estas indicaciones no tienen porqué ser válidas en todas las situaciones, pero se cumplen de forma general. En la Tabla 9.3 se proporcionan las características técnicas más relevantes de algunos robots comerciales.

9.3. SEGURIDAD EN INSTALACIONES ROBOTIZADAS

El tema de la seguridad y prevención de accidentes en los sistemas robotizados, suele tratarse de manera marginal en casi todos los textos y demás fuentes de información que sobre robots industriales existen. Sin embargo, éste es un aspecto crítico durante el desarrollo y explotación de una célula robotizada. Las consideraciones sobre la seguridad del sistema robotizado cobran especial importancia fundamentalmente por dos razones. En primer lugar, por el motivo intrínseco de que el robot, como se analizará más adelante, posee mayor índice de riesgo a un accidente que otra máquina de características similares. En segundo lugar, por un aspecto de aceptación social del robot dentro de la fábrica, aceptación difícil por lo general hoy en día.

La realidad, sin embargo, es que el número de accidentes ocasionados por los robots industriales no es ni mucho menos alarmante, existiendo pocos datos al respecto, siendo pocos los países que cuentan con suficiente información al respecto.

9.3.1. Causas de accidentes

Para prevenir los posibles accidentes ocasionados por los robots, hay que comenzar detectando *qué* tipo de accidentes se producen, para después analizar el *porqué* se originan y determinar *cómo* pueden evitarse.

En principio, y dado lo similar de sus características, los riesgos de accidente en un entorno de trabajo con robots industriales deberían ser similares a los debidos al empleo de máquinas herramientas con control numérico. Sin embargo, hay una serie de circunstancias que aumentan el nivel de riesgo en el caso de los robots. Estas causas se reflejan de manera comparativa en la Tabla 9.4. Además, el hecho de que el robot trabaje en muchas ocasiones en ambientes de alto riesgo de accidente (forjas, prensas, pintura, etc.) contribuye a aumentar la probabilidad y gravedad del accidente.

Los tipos de accidentes causados por robots industriales, además de los ocasionados por causas tradicionales (electrocuciones al instalar o reparar el equipo, quemaduras, etc.), son debidos a:

- Colisión entre robots y hombre.
- Aplastamiento al quedar atrapado el hombre entre el robot y algún elemento fijo.
- Proyección de una pieza o material (metal fundido, corrosivo) transportada por el robot.

Estos causas son agravadas por la gran velocidad con la que los robots pueden realizar sus movimientos, además de su elevada energía estática y dinámica.

Establecidos los tipos principales de accidentes, es preciso localizar cuáles son las causas que los originan. Los accidentes provocados por robots industriales se deben normalmente a:

- Un mal funcionamiento del sistema de control (software, hardware, sistemas de potencia).
- Acceso indebido de personal a la zona de trabajo del robot.
- Errores humanos de los operarios en las etapas de mantenimiento, programación, etc.
- Rotura de partes mecánicas por corrosión o fatiga.
- Liberación de energía almacenada (eléctrica, hidráulica, potencial, etc.).
- Sobrecarga del robot (manejo de cargas excesivas).
- Medio ambiente o herramienta peligrosa (láser, corte por chorro de agua, oxicorte, etc.).

Tabla 9.4. Comparación entre un robot industrial y otras máquinas de control numérico

Robot industrial	Máquina convencional de control numérico
Movimiento simultáneo de varios ejes (seis o más).	Normalmente mueve simultáneamente sólo uno o dos ejes.
Movimiento independiente de cada uno de los ejes. Trayectorias complejas.	Movimientos limitados y con trayectorias simples.
Campo de acción fuera del volumen cubierto por la propia máquina. Espacio de trabajo no reconocido fácilmente.	Campo de acción dentro del volumen abarcado por la máquina.
Campo de acción solapado con el de otras máquinas y dispositivos.	Normalmente sin solapamiento del campo de acción.

9.3.2. Medidas de seguridad

Una vez vistos los tipos de accidentes y las causas que los originan, es preciso determinar las medidas de seguridad a seguir que disminuyan el riesgo y la gravedad.

Es importante considerar que según estudios realizados por el *Instituto de Investigaciones de Seguridad en el Trabajo* de Tokyo, el 90% de los accidentes en líneas robotizadas ocurren durante las operaciones de mantenimiento, ajuste, programación, etc., mientras que sólo el 10% ocurre durante el funcionamiento normal de la línea.

Este dato es de gran relevancia y pone de manifiesto la gran importancia que tiene, para lograr un nivel de seguridad adecuado, el impedir el acceso de operarios al campo de acción del robot mientras éste está en funcionamiento.

La seguridad en sistemas robotizados presenta, por tanto, dos vertientes: aquella que se refiere a la seguridad intrínseca al robot y que es responsabilidad del fabricante; y aquella que tiene que ver con el diseño e implantación del sistema y su posterior utilización, programación y mantenimiento, responsabilidad del usuario.

En este sentido, se ha desarrollado la normativa europea EN 775, adoptada en España como norma UNE-EN 775 de título "Robot manipuladores. Seguridad", que además de proporcionar a diseñadores y fabricantes un marco de trabajo que les ayude a producir máquinas seguras en su utilización, presenta una estrategia de trabajo para el desarollo y selección de medidas de seguridad. Esta estrategia comprende las siguientes consideraciones:

- Determinación de los límites del sistema: intención de uso, espacio y tiempos de trabajo, etc.
- Identificación y descripción de todos aquellos peligros que pueda generar la máquina durante las fase de trabajo. Se deben incluir los riesgos derivados de un trabajo conjunto entre la máquina y el operador y los riesgos derivados de un mal uso de la máquina.
- Definición del riesgo de que se produzca el accidente. Se definirá probabilísticamente en función del daño físico que pueda producir.
- Comprobar que las medidas de seguridad son adecuadas.

La selección de las medidas de seguridad viene definida por las siguientes consideraciones.

Medidas de seguridad a tomar en la fase de diseño del robot

En el diseño del robot y de su sistema de control debe considerarse siempre el posible accidente, tomándose las acciones oportunas para evitarlo en la medida de lo posible. Así, el propio robot debe contar con una serie de medidas internas encaminadas a evitar posibles accidentes:

- **Supervisión del sistema de control**: El sistema de control debe realizar una continua supervisión del correcto funcionamiento de todos los subsistemas (bucles de realimentación, accionamientos, etc.) y de incluso él mismo (*watch-dog*).
- **Paradas de emergencia**: Deben disponerse paradas de emergencia que desenergicen completamente al robot.
- **Velocidad máxima limitada**: El sistema de control asegurará que la velocidad máxima de los movimientos cuando una persona se encuentra en las proximidades del robot (fase de programación por ejemplo) sea inferior a la nominal (como referencia debe ser inferior a 0,3 m/s).
- **Detectores de sobreesfuerzo**: Se incluirán detectores de sobreesfuerzo en los accionamientos que los desactiven cuando se sobrepase un valor excesivo (caso de colisión o de atrapar a una persona contra una parte fija).
- **Pulsador de seguridad**: Las paletas y consolas de programación dispondrán de un dispositivo de seguridad (pulsador de hombre muerto) que impidan el movimiento accidental del robot.
- **Códigos de acceso**: El acceso a la unidad de control y el arranque, parada y modificación del programa, estarán limitadas mediante el empleo de llaves, códigos de seguridad, etc.

- **Frenos mecánicos adicionales**: Si el robot maneja grandes cargas, se deberán de incluir frenos mecánicos que entren en funcionamiento cuando se corte la alimentación de los accionadores. Asimismo, se dispondrán medios para desactivar estos frenos de forma manual.
- **Comprobación de señales de autodiagnóstico** en la unidad de control previamente al primer funcionamiento (niveles de tensión de las fuentes de alimentación, leds indicadores, mensajes de error, etc.).

Medidas de seguridad a tomar en la fase de diseño de la célula robotizada

En el establecimiento del *lay-out* de la célula se ha de considerar la utilización de barreras de acceso y protecciones en general que intenten minimizar el riesgo de aparición de un accidente. De forma general se pueden citar las siguientes:

- **Barreras de acceso a la célula**: Se dispondrán barreras en torno a la célula, que impidan el acceso a personas (parada inmediata al entrar en la zona de trabajo) (Figura 9.9).
- **Dispositivos de intercambio de piezas**: En caso de que el operador deba poner/recoger piezas situadas dentro del área de trabajo del robot, se utilizarán dispositivos que permitan realizar estas acciones a distancia, utilizando, por ejemplo, mesas giratorias (Figura 9.10).

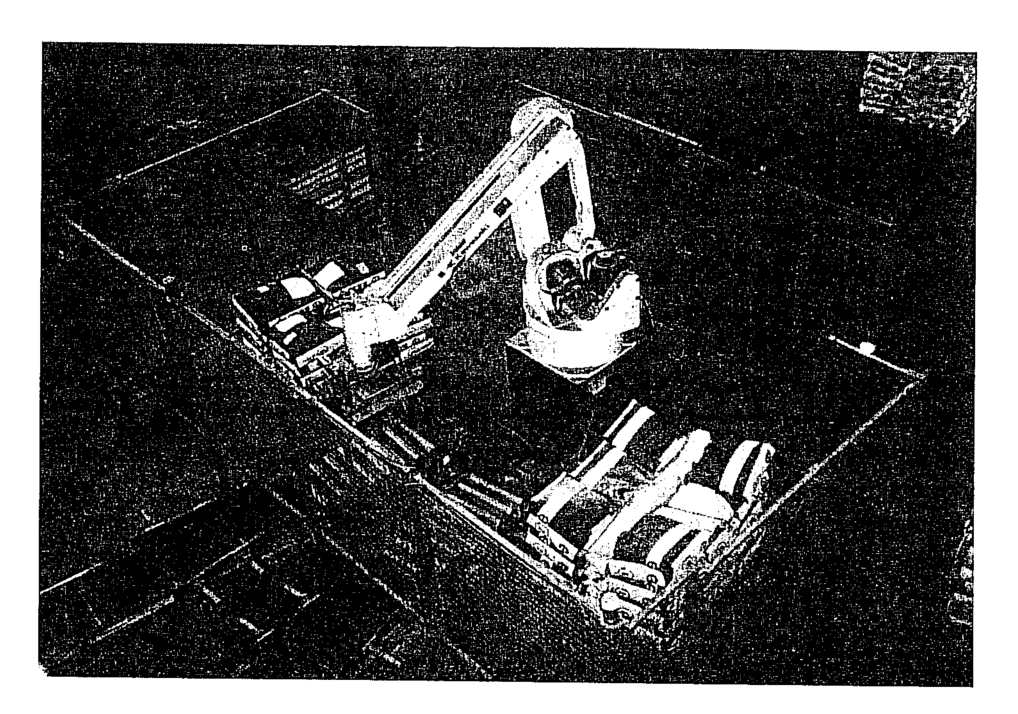


Figura 9.9. Célula robotizada de paletizado con vallas de protección para impedir el acceso. (Cortesía de INSER Robótica.)

- **Movimientos condicionados**: En el caso de que durante el funcionamiento de la célula el operario deba entrar en determinados momentos dentro del campo de acción del robot (para alimentar de nuevas piezas al robot por ejemplo), se programará a éste de manera que no efectúe movimiento alguno durante estos instantes.
- **Zonas de reparación**: Se preverá la existencia de zonas de reparación y mantenimiento. Estas zonas, dentro del campo de acción del robot, estarán fuera de su zona de trabajo. En ellas se asegurará mediante diferentes dispositivos que el robot no realizará movimientos de manera automática.
- **Condiciones adecuadas en la instalación auxiliar**: Sistema eléctrico con protecciones, aislamientos, etc., sistemas neumáticos o hidráulicos correctos.

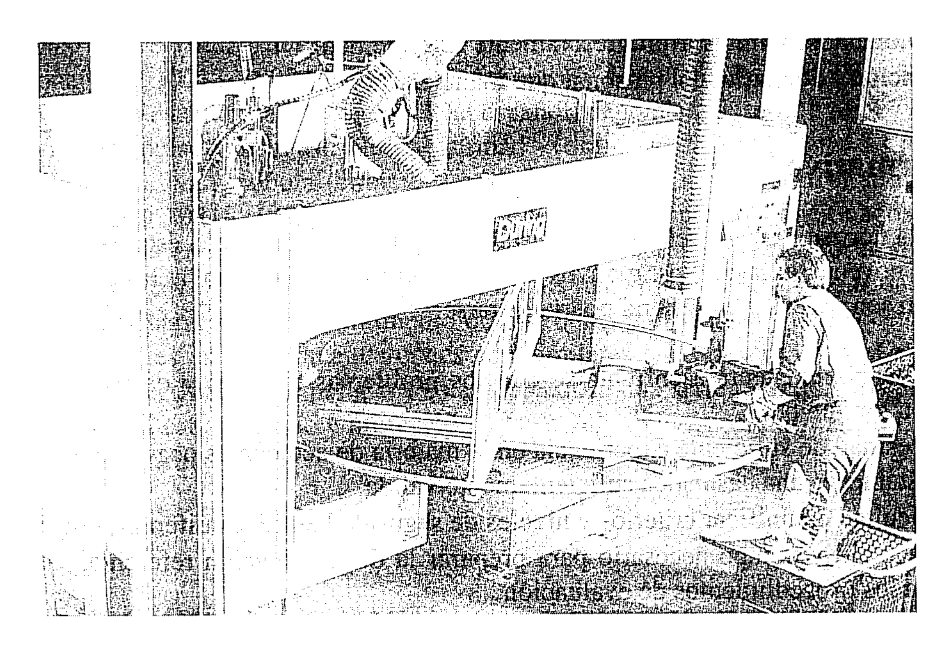


Figura 9.10. Operador utilizando una mesa giratoria para la alimentación de piezas en una célula robotizada de soldadura. (Cortesía de KUKA.)

Medidas de seguridad a tomar en la fase de instalación y explotación del sistema

Por otra parte, durante la utilización del sistema y en especial durante las fases de instalación y puesta en marcha, deben respetarse rigurosamente determinadas normas que reducirán el riesgo de accidente. Además, es importante que exista información en la propia planta de la posibilidad de esos riesgos, así como que los operarios tengan la formación adecuada. Estas consideraciones se pueden resumir de forma breve en los siguientes puntos.

- **Abstenerse de entrar en la zona de trabajo**: Durante la programación e implantación de la aplicación, se procurará permanecer, dentro de lo posible, fuera del campo de acción del robot. Éste trabajará a velocidades lentas. En cualquier caso, se deberá salir fuera del área de trabajo

cuando el robot vaya a trabajar de manera automática, aun siendo en fase de pruebas. Es también aconsejable que, siempre que sea posible, la fase de programación se realice con dos operarios, observando uno de ellos la marcha del proceso, estando dispuesto a accionar el paro de emergencia en caso de necesidad.

- **Señalización adecuada**: La célula estará dotada de una adecuada señalización del estado del robot o línea robotizada mediante señales luminosas y acústicas. Así, será aconsejable que antes de comenzar el ciclo de trabajo tras una parada se avise mediante una sirena e indicación luminosa. Del mismo modo, señales luminosas llamarán la atención sobre el hecho de que el robot está funcionando y su riesgo potencial de accidente.
- **Prueba progresiva del programa del robot**: El desarrollo y ejecución del programa del robot, y de toda la célula en sí, deberá hacerse con sumo cuidado. El programa deberá ejecutarse primeramente a velocidad lenta y paso a paso. A continuación se podrá ejecutar de manera continua, pudiéndose aumentar progresivamente la velocidad.
- **Formación adecuada** del personal que manejará la planta.

9.3.3. Normativa legal

En cuanto a la normativa legal relativa a la instalación y empleo de robots, ésta ha sido hasta principios de los años noventa escasa. Los motivos que han llevado al retraso a la hora de proponer una normalización son varios, y en general podrían citarse:

- La tendencia existente a enfrentarse con los problemas técnicos y de mercado (ventas) antes que con ningún otro.
- La necesidad de suficiente experiencia en la materia de accidentes ocasionados por robots como para establecer una casuística suficientemente válida.
- La dificultad en unificar criterios y niveles de seguridad entre los diferentes usuarios y países.
- La dificultad y tiempo necesario para preparar la documentación referente a la normativa, así como a los procedimientos de evaluación.

En la actualidad, la normativa más relevante existente al respecto a nivel mundial es la siguiente:

Normativa internacional ISO 10218 :1992

Se trata de una normativa realizada por el organismo internacional de estandarización [ISO-92]. Es relativamente reciente, pues data del año 1992. A grandes rasgos contiene la siguiente información: una sección sobre el análisis de la seguridad, la definición de riesgos y la identificación de posibles fuentes de peligros o accidentes. Contiene además una sección sobre diseño y fabricación, que dedica un breve análisis al diseño de sistemas robotizados, teniendo en cuenta aspectos mecánicos, ergonómicos y de control.

La mayoría de las indicaciones que se proporcionan son de carácter general. Hay que mencionar que dado el corto espacio de tiempo que lleva vigente, no se conocen datos fiables sobre los resultados derivados de su utilización.

Normativa americana ANSI/RIA R15.06-1992

Se trata de una normativa realizada por el *Instituto Nacional de Normalización de Estados Unidos* (ANSI) [ANSI-92]. También data del año 1992, siendo una revisión de una normativa publicada en 1986.

Es relativamente breve, pero presenta algunas características que merecen destacarse. Por ejemplo, la inclusión, en el apartado sobre la definición de riesgos, de algunos epígrafes que versan sobre la probabilidad de la aparición de un accidente y la severidad del posible daño físico a una persona, dependientes del nivel de experiencia del operador y la frecuencia en la que éste se encuentra en zona de peligro.

Normativa europea EN 775 y española UNE-EN 775

El *Comité Europeo de Normalización* (CEN) aprobó en 1992 la normativa EN 775, adaptación de la norma internacional ISO 10218 :1992. A su vez la *Asociación Española de Normalización* (AENOR) adoptó en marzo de 1996 esta normativa (traducida al español) denominándose UNE-EN 775 y que lleva por título : "Robots manipuladores industriales. Seguridad" [UNE-96].

Por lo tanto, la normativa española (coincidente con la europea) se ha desarrollado sobre la normativa internacional ISO 10218 :1992, incorporando y adaptando aspectos relacionados en la norma europea "EN 292:1991 Safety of Machinery - Basic concepts, general principles for design" [EN-91].

Esta normativa UNE-EN 775 incluye una serie de requisitos para mejorar la seguridad en las fases de diseño, utilización, reparación y mantenimiento de los robots industriales y de las células robotizadas. En términos generales estas recomendaciones son similares a las descritas en el epígrafe 9.3.2, enumerándose las posibles fuentes de peligro y estableciéndose una serie de requisitos en las fases de diseño y construcción del robot, diseño y protección de la célula robotizada y utilización, instalación y puesta en marcha de la misma.

9.4. JUSTIFICACIÓN ECONÓMICA

Es norma habitual antes de llevar a cabo cualquier proyecto de ingeniería, el realizar un análisis económico del mismo; análisis que determinará de forma aproximada su viabilidad y rentabilidad. Cuando además el proyecto consiste en su mayor parte en la inclusión de material tecnológicamente avanzado o en la automatización de un proceso ya conocido, este análisis se convierte en imprescindible. Este es el caso de un proyecto de ingeniería en el que se utiliza como equipo principal un robot industrial.

Los beneficios sociales o lo avanzado de la tecnología a utilizar, aunque se consideren cuestiones importantes, son normalmente aspectos secundarios, pues el primer objetivo de cualquier empresa que acomete un proyecto de estas características es obtener de él un rendimiento económico, ya sea a corto o a largo plazo.

En este epígrafe se va a considerar la metodología a aplicar para la realización de este análisis. Esta metodología se basa en el análisis de los factores que la aplicación de un robot trae consigo.

9.4.1. Factores económicos y datos básicos necesarios

Para poder realizar el análisis económico de un proyecto con robot es necesario tener en cuenta un conjunto de factores económicos particulares, además de contar con una información previa sobre algunos datos básicos del proyecto. Como factor económico principal se considera el tipo de instalación a desarrollar, y como datos básicos se pueden citar los siguientes: coste de inversión, costes de instalación y ahorros o beneficios resultantes del proyecto. Todo ello teniendo en cuenta al robot de manera directa.

Tipo de instalación

A la hora de realizar un proyecto con robot existen dos posibles situaciones. Primero, que la aplicación que se quiere realizar sea de marcado carácter novedoso. En este caso, la inclusión de un robot es sólo una posibilidad entre diversas alternativas. Se deberán considerar todas y seleccionar la que resulte más beneficiosa según los criterios de inversión de la compañía. En este tipo de proyectos existe una dificultad añadida, pues al tratarse de una nueva aplicación muchos datos sobre el proceso no son conocidos de manera fiable, pudiéndose llegar a deducir conclusiones equívocas. Puede llegar a ser ciertamente compleja la evaluación económica y técnica en este tipo de proyectos si no se posee un conocimiento adecuado sobre la implantación de células robotizadas.

El segundo caso, de aplicación más frecuente, consiste en la automatización de un método de operación que se realiza en buena medida de forma manual, con la inclusión de un robot. En esta situación el análisis resulta más sencillo, pues los parámetros y tiempos de operación manual o robotizada son fácilmente comparables. Se podrá asimismo realizar una comparación de costes y beneficios, encontrándose todos los factores de producción perfectamente delimitados.

Con independencia de cuál sea la situación en la que se encuentre es necesario, para realizar el análisis, utilizar lo que se denomina el análisis económico.

Datos básicos de coste

Al igual que en cualquier proyecto de ingeniería, los datos básicos de coste a tener en cuenta se dividen en dos categorías: costes de inversión y costes de explotación. Se estudian a continuación dichos costes, orientados siempre hacia el caso particular de una célula robotizada.

Costes de inversión: Los costes de inversión son aquellos costes necesarios para realizar la implantación física del sistema. Como principales costes de inversión se incluyen los siguientes.

- *Coste de adquisición del robot*: Este coste incluye el robot equipado con aquellas características que se consideren oportunas. No se incluirá aquí la herramienta a colocar en el extremo del robot. El coste del robot es muy variable, y depende principalmente de las prestaciones que sea capaz de suministrar. Por regla general, robots más caros podrán adaptarse a un mayor número de tareas diferentes, mientras que robots de un precio inferior se adaptarán peor y se tendrán que desarrollar herramientas especiales para ellos.
- *Coste de herramientas y equipos especiales*: Se incluyen en este apartado tanto la herramienta o efector final a utilizar por el robot, como todos aquellos equipos especiales que se hayan de

utilizar debido a la inclusión del robot. Este coste va íntimamente ligado al coste de compra del robot.

- *Coste de ingeniería e instalación*: Aquí se deben incluir en principio aquellos costes derivados del estudio de planificación y diseño, así como costes de desarrollo para la utilización del robot. Posteriormente se han de añadir aquellos costes específicos de la instalación del sistema. Habrá que tener en cuenta de forma especial aquellos costes generados de forma directa por la inclusión del robot, para poder así comparar, cuando sea necesario, con los costes de índole similar utilizando soluciones alternativas al robot.

Costes y beneficios de explotación: Son aquellos costes derivados de la utilización directa de la célula. También se ha de tener en cuenta el ahorro o beneficios que la utilización del robot, o la automatización de forma general, comporte. Entre los costes de explotación y ahorro se pueden mencionar los siguientes:

- *Mano de obra:* Como costes se deberán incluir tanto la mano de obra directa de operación de la planta, como la mano de obra que de forma indirecta interviene en la explotación, como puede ser en la supervisión, planificación, etc. Se debe computar como un ahorro la mano de obra sustituida por la automatización, siempre teniendo en cuenta los costes sociales que pudieran generarse.
- *Mantenimiento*: Se incluyen aquí los costes derivados del mantenimiento de los equipos de la planta, exceptuando la mano de obra considerada en el factor anterior. Entre los equipos es necesario destacar el mantenimiento del robot. Estos datos sobre reparaciones, repuestos, etc., han de suponerse en su mayor parte. Habrá que ser consecuente y tener en cuenta las demandas o requerimientos que el trabajo y su entorno exijan al robot y a los demás equipos.
- *Aumento de calidad e incremento de productividad*: Se trata de beneficios importantes a tener en cuenta como consecuencia directa de la utilización de un robot. El aumento de calidad se debe principalmente a que el robot realiza su tarea de manera uniforme y consistente, sin la aparición de altibajos propios del estado de ánimo de un operario. El robot puede, en la mayoría de los casos, trabajar tan rápido como un operario, pero además puede seguir realizándolo durante varios turnos sin problema alguno.
- *Entrenamiento*: Al tratarse de una tecnología relativamente moderna, será necesario formar previamente a la mano de obra y al personal que de alguna manera intervenga en la célula. Este entrenamiento ha de ser continuo y por ello se incluye como gasto de explotación y no como gasto de inversión.

9.4.2. El robot como elemento principal del análisis económico

El robot desde el punto de vista económico presenta unas características completamente distintas a los demás equipos de una planta, y como tal merece ser tratado de forma independiente.

Ya se han mencionado en el epígrafe anterior algunas de esas características, como puede ser su coste, mantenimiento o el aumento de calidad y productividad que su presencia aporta. Sin embargo, no se ha tenido en cuenta uno de los factores que definen de por sí a un robot industrial: la versatilidad o flexibilidad de su uso.

Normalmente, en una aplicación industrial se considera que la vida de los equipos de fabricación se corresponde con la del producto que se está fabricando. Es decir, cuando se ha de cambiar de producto, ya sea por cuestiones de mercado o necesidades técnicas, las máquinas empleadas se quedan obsoletas o su reciclaje para el nuevo proceso resulta muy costoso. Su rendimiento posterior al ciclo de vida del producto es prácticamente nulo.

Un robot, en cambio, es una herramienta programable y flexible y por tanto susceptible de adaptarse, casi sin coste alguno para ser utilizado en diversas aplicaciones. Esta característica es sumamente interesante, pues implica que el robot, como equipo de fabricación, no está sujeto al ciclo de vida del producto, sino que posee un ciclo de vida propio y por tanto su rentabilidad puede ser mayor. Existen, sin embargo, dos matices que se deben mencionar.

Primero, aunque el robot sea reprogramable nunca se asegura que tras concluir el ciclo del producto al que contribuía a fabricar, el robot pueda incluirse de forma eficiente en un nuevo proceso de fabricación. Existirán muchas más posibilidades que con cualquier otra herramienta, pero no la seguridad total. Segundo, se tiende a utilizar el robot, precisamente por su flexibilidad, en procesos de fabricación con ciclos de vida cortos. Lo que explica que para la primera aplicación sea complicado justificar su inversión.

Esto lleva a tener que considerar un valor de recuperación o retorno del coste del robot al finalizar cada uno de los proyectos en los que se aplica. Lógicamente, este valor de retorno deberá ir disminuyendo con cada proyecto, ya sea por su deterioro o por su obsolescencia. Este valor de retorno es muy importante y se concretará en el siguiente epígrafe cuando se hable de los métodos de análisis económico.

Otro factor a tener en cuenta, ya mencionado en el epígrafe anterior, es el aumento de productividad que el robot aporta. Aumento que es consecuencia directa de la realización del trabajo de forma más rápida y con menores interrupciones (trabajo en varios turnos seguidos). Este factor se verá con más detalle en el siguiente epígrafe.

9.4.3. Métodos de análisis económico

Existen diversos métodos aplicables para realizar el análisis económico de una inversión y poder decidir si ésta va a resultar rentable o no. Se trata de métodos de aplicación general a cualquier tipo de inversión o proyecto [MOCHÓN-90] [GROOVER-89], aunque aquí se particularizarán para proyectos que incluyan la utilización de robots. Los tres métodos a estudiar son:

- Período de Recuperación.
- Método del Valor Actual Neto (VAN).
- Método de la Tasa Interna de la Renta (TIR).

Cada uno de los tres métodos es de aplicación independiente y sirven para realizar análisis similares. Teóricamente, de la aplicación de los tres a un caso concreto se deberían extraer las mismas conclusiones. Es preciso considerar que al basar gran parte de los razonamientos en datos supuestos, los resultados, en ocasiones pueden ser de validez limitada.

En este epígrafe se presenta una breve descripción de cada uno, sin querer llegar a profundizar demasiado en ellos. Además, cada empresa suele desarrollar sus propios procedimientos, donde realiza una evaluación particular del proyecto y valora los resultados según sus propios criterios.

Período de recuperación

Se entiende como período de recuperación a aquel intervalo de tiempo que transcurre desde el comienzo del proyecto hasta que el flujo de caja (*cash-flow*) neto o acumulado sea mayor que cero. Es decir, indica cuándo se empieza a tener ingresos positivos sumando todas las entradas y salidas de capital desde el comienzo del proyecto. Como es lógico, se busca un período de recuperación lo más corto posible, pues transcurrido este tiempo es cuando realmente la inversión comienza a ser rentable. Períodos de recuperación cercanos al ciclo de vida del proyecto o mayores suponen un descarte automático de la inversión, pues en un caso los riesgos no merecen la pena, y en el otro nunca se recuperará lo invertido.

El período de recuperación se mide normalmente en años, y viene dado por el valor de n que cumple la ecuación:

$$\sum_{j=0}^{n}\left(R_j - C_j\right) = 0 \qquad [9.\ 1]$$

en la que R_j y C_j representan los ingresos y costes estimados durante el transcurso del año j. Lógicamente los costes de inversión vendrán incluidos en el valor de C_0 siendo $R_0 = 0$. Para simplificar se supone un *cash-flow* constante durante la vida del proyecto y despejar n de la ecuación:

$$-\mathrm{CI} + n(R - C) = 0 \qquad [9.\ 2]$$

en la que CI representa el coste de la inversión inicial y (R-C) el cash-flow anual. Despejando n resultará:

$$n = \frac{\mathrm{CI}}{R - C} \qquad [9.\ 3]$$

La realización de un proyecto con robot no cambia significativamente la aplicación de este método. Únicamente considerar el aumento de productividad y por tanto de ingresos que su utilización acarrearía. Se muestra a continuación un ejemplo.

Ejemplo 9.1.

Supóngase un proyecto que cumpla los datos mostrados en la Tabla 9.5. La influencia de utilizar un robot se notará en los ingresos que se esperan recibir cada año, función directa de la productividad. Para hallar el período de recuperación bastará aplicar la ecuación [9.1], y es sencillo obtener que el período de recuperación n ha de tener un valor de 2.8 años para cumplirla.

$$-200 + (80 - 30) + (120 - 50) + 0.8(150 - 80) = 0$$

Método del Valor Actual Neto (VAN)

El método anterior, aunque útil por su simplicidad, presenta el inconveniente de que no se considera el valor temporal del dinero. En el método del VAN este concepto sí se tiene en cuenta. Se considera que los elementos de coste de la inversión incluyen además del precio del material a adquirir, el tipo de interés a abonar por financiación externa. Se consideran además los beneficios como rendimientos futuros netos que se espera que obtener de la inversión inicial.

Se considera entonces cómo VAN de la inversión al valor presente del *cash-flow* futuro, utilizando como tasa de descuento del coste el tipo de interés. Es decir, el VAN podrá expresarse mediante la siguiente formula:

$$VAN = \sum_{j=0}^{n} \frac{\left(R_j - C_j\right)}{(1+i)^j} \qquad [9.4]$$

donde n es el período de vida del proyecto, R_j y C_j los ingresos y gastos estimados durante el año j, e i es el tipo de interés o rentabilidad. Lógicamente, un valor de VAN positivo indica una inversión rentable, y cuanto mayor sea aquél mayor será la rentabilidad. Por el contrario, un valor del VAN negativo supondrá una inversión no rentable.

Como valor de i se elegirá aquel que la empresa considere como aceptable para asumir el riesgo. En este método si es posible considerar el valor del retorno del robot. Si se supone que al final del proyecto el valor de retorno del robot puede considerarse como M', el VAN quedaría:

$$VAN = \sum_{j=0}^{n} \frac{\left(R_j - C_j\right)}{(1+i)^j} + \frac{M'}{(1+i)^{n+1}} \qquad [9.5]$$

Se hace notar que el valor de retorno del robot se considera al final del último año del proyecto. Se aplica a continuación sobre un ejemplo.

Tabla 9.5. Datos económicos de un proyecto (en millones de pesetas)

Período de vida	5 años					
Años	0	1	2	3	4	5
Coste de inversión	200	-	-	-	-	-
Costes de explotación	-	30	50	80	80	80
Ingresos	-	80	120	150	150	150

Ejemplo 9.2.

Se consideran los datos de un proyecto dados por la Tabla 9.5. Suponiendo que el valor de depreciación del robot es lineal con respecto al paso del tiempo y conociendo la vida útil del robot y su coste inicial, es inmediato calcular el valor de retorno del mismo tras el quinto año. Suponiendo este valor de 5 millones, y que la rentabilidad que se quiere obtener es del 0.2, el VAN vendrá dado por la siguiente ecuación:

$$\mathrm{VAN} = -200 + \frac{80-30}{(1+0.2)} + \frac{120-50}{(1+0.2)^2} + \frac{150-80}{(1+0.2)^3} + \frac{150-80}{(1+0.2)^4} + \frac{150-80}{(1+0.2)^5} + \frac{5}{(1+0.2)^6}$$

y operando se obtiene un valor del VAN de -5.65, lo que indica que el proyecto no es rentable, es decir, que no llega a proporcionar el 0.2 de rentabilidad propuesto.

Método de la Tasa Interna de la Renta (TIR)

Este método es muy similar al del VAN, aunque su enfoque es ligeramente distinto. Lo que se trata de hallar es la tasa de rendimiento que se va a obtener del proyecto y por tanto de la inversión. Si en el método del VAN se suponía una tasa de rendimiento aceptable, con el TIR se trata de ver cuál es esa tasa de rendimiento máxima posible. Para ello se utiliza la fórmula del VAN, pero considerando que éste es nulo y que la tasa de rentabilidad i es desconocida. De esta ecuación se obtendrá i, y en base a ella se decidirá si merece la pena el riesgo que supone la inversión. En todo lo demás es similar al método del VAN.

Ejemplo 9.3.

Se consideran los mismo datos de partida que en el Ejemplo 9.2. Sin embargo, se desconoce ahora el valor de la rentabilidad, que es precisamente lo que se quiere obtener. Aplicando la ecuación [9.5] y considerando el valor del VAN igual a cero, se tiene:

$$0 = -200 + \frac{80-30}{(1+\mathrm{i})} + \frac{120-50}{(1+\mathrm{i})^2} + \frac{150-80}{(1+\mathrm{i})^3} + \frac{150-80}{(1+\mathrm{i})^4} + \frac{150-80}{(1+\mathrm{i})^5} + \frac{5}{(1+\mathrm{i})^6}$$

Despejando i se obtiene el valor del TIR, que en este caso es de 0.187.

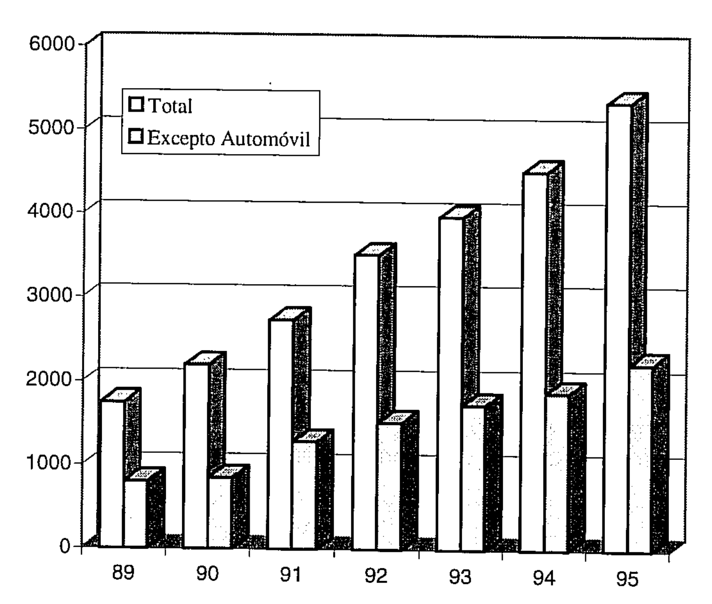


Figura 9.11. Evolución del parque de robots en España hasta el año 1995 (Fuente: AER).

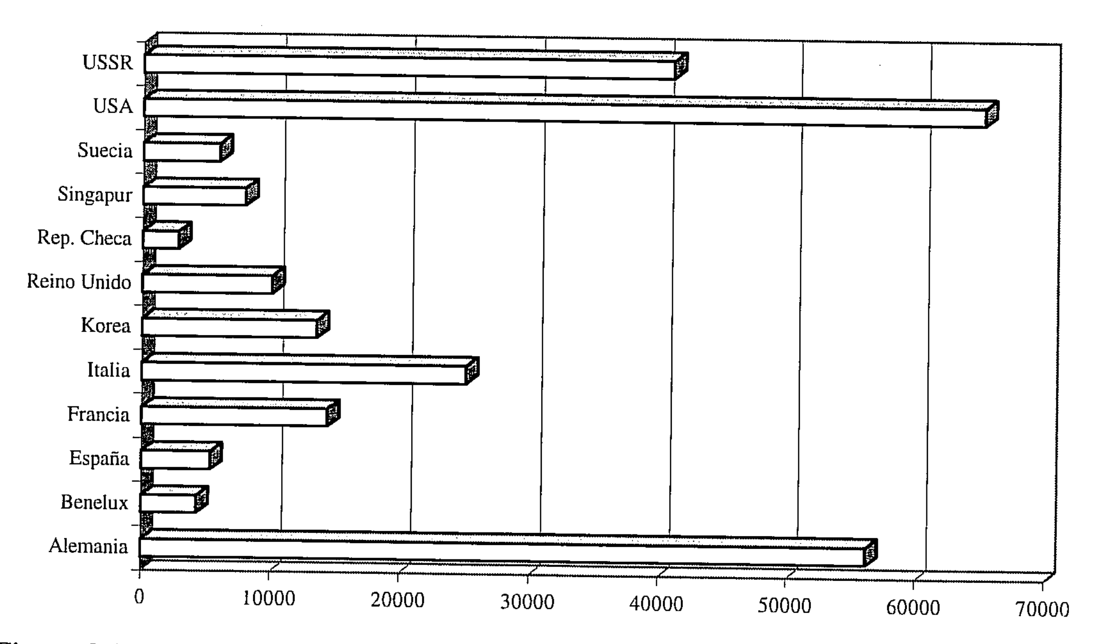


Figura 9.12. Parque mundial (excepto Japón) de robots a comienzos del año 1996 (Fuente: AER).

9.5. MERCADO DE ROBOTS

Por muchas de las consideraciones realizadas a lo largo de este capítulo, la instalación y posterior explotación de un robot industrial se adapta mejor a un tipo de industria que a otro. Esto es debido principalmente a que determinadas aplicaciones propias de estas industrias permiten la utilización de robots de forma muy rentable. Tradicionalmente, cuando se presentan estadísticas sobre implantación de robots en una determinada zona, se suele distinguir entre la distribución de robots por sectores y por aplicaciones. La propia *Federación Internacional de Robótica* (IFR) propone una serie de indicaciones sobre cómo realizar esta clasificación.

La realización de este tipo de estadísticas no siempre es sencilla, pues suelen aparecer contradicciones entre los datos proporcionados por suministradores y usuarios, cada uno atendiendo a sus propios intereses. Además, en numerosas ocasiones, la asignación de un robot a un sector o

Tabla 9.6. Densidad de robots e índice de paro de varios países en 1996 (Fuente: AER)

Países	Densidad de robots	Índice de paro	Número de robots
Japón	250	3 %	413.578
Suecia	60	9 %	5.911
Alemania	58	9 %	56.175
Italia	55	12 %	25.096
Estados Unidos	35	7 %	65.198
Francia	33	12 %	14.376
España	22	22 %	5.346

Densidad de robots = nº de robots cada 10.000 trabajadores de la industria

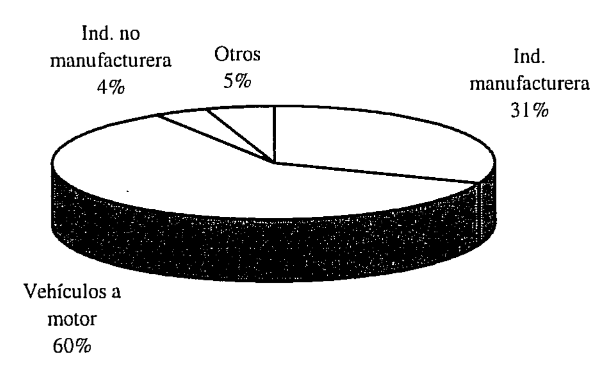


Figura 9.13. Distribución de robots en España por sectores a comienzos del año 1996 (Fuente: AER).

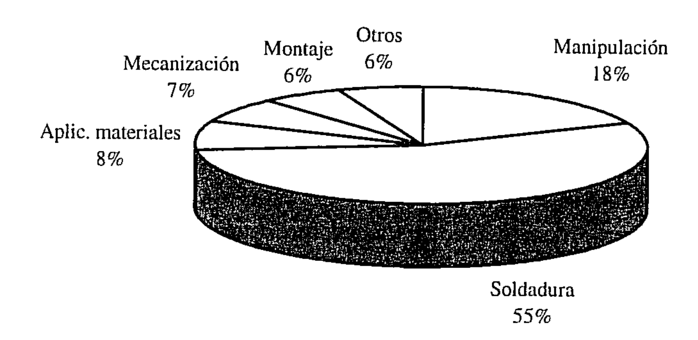


Figura 9.14. Distribución de robots por aplicaciones en España a comienzos del año 1996 (Fuente: AER).

aplicación no está lo suficientemente clara. Aun así, estos datos son extremadamente útiles para conocer tendencias y la situación aproximada del mercado.

Los datos estadísticos sobre el mercado de robots son actualizados anualmente por la IFR y publicados en cada país socio. En España, la *Asociación Española de Robótica* (AER), como organismo miembro de la IFR, publica anualmente estos datos. Las estadísticas presentadas en este epígrafe han sido extraídos del informe anual correspondiente al año 1996 y que incluye datos a fecha de enero de 1996 [AER-96].

España, aunque no se trate de un país con un gran índice de robotización, sí presenta la particularidad de albergar numerosas industrias de automoción, lo que, como más adelante se verá, condiciona de forma importante las características de su parque de robots. Normalmente, el índice de robotización de un país se define en función del número de robots en funcionamiento por cada 10.000 trabajadores en la industria.

En la Figura 9.11 se muestra la evolución seguida por el parque de robots en España durante los últimos años. Se aprecia un crecimiento constante y sostenido, que se espera se mantenga durante algún tiempo, pues la robótica es aún una tecnología joven, susceptible de abrirse camino en aplicaciones y sectores nuevos, como puede ser el de la construcción. En total el número de robots funcionando en nuestro país durante el año 1995 alcanzó los 5.346.

Al año, se están vendiendo en todo el mundo unas 75.500 unidades de robots industriales. Alemania y Suecia destacan, a excepción de Japón, como los países con más número de robots implantados por cada 10.000 trabajadores de la industria (densidad de robots), aunque en cuanto al número absoluto de robots en sus fábricas se encuentren por debajo de Estados Unidos o Japón (Figura 9.12). Este último (no incluido en la figura) tiene la particularidad de incluir en su parque de robots toda una gama de manipuladores de propósito específico que no son considerados como tal en el resto de los países y que llegan a la desorbitante cifra de 413.578 para el año 1995. La Tabla 9.6 muestra los datos correspondientes a la densidad de robots en algunos países industrializados, y su comparación con el índice aproximado de paro en los mismos.

La Tabla 9.8, junto con la Figura 9.13, muestra la distribución de robots por sectores en España a principios del año 1996. Destaca de forma apreciable la cuota de mercado que el sector dedicado a vehículos a motor, es decir, la industria del automóvil, posee en España.

Tabla 9.7. Distribución de robots en España por aplicaciones (1996) (Fuente: AER)

Aplicación	Hasta 1994	Alt. 95	Baj. 95	Total
Aplicación adhesivos	152	28	0	180
Apliacación materiales	80	0	0	80
Carga y descarga de máquinas	282	32	0	314
Corte mecánico, desbarbado	40	7	0	47
Corte por láser	1	1	0	2
Corte por chorro de agua	50	9	0	59
Desmoldeo en plástico	175	114	0	289
Enseñanza, investigación	108	19	0	127
Inserción y montaje	24	0	0	24
Manipulación de materiales	379	71	0	450
Manipulación en fundición	13	28	0	41
Manipulación para paletización	99	23	0	122
Medición e inspección	26	0	0	26
Montaje mecánico	183	0	0	183
Pintura	128	29	0	157
Soldadura al arco	1004	156	0	1160
Soldadura por puntos	1480	244	0	1724
Soldadura, otros	63	0	0	63
Varios	229	69	0	298
Total	4516	830	0	5346

Tabla 9.8. Distribución de robots en España por sectores (1996) (Fuente: AER)

Sector	Hasta 1995	Alt. 95	Baj. 95	Total
Alimentación y bebidas	18	6	0	24
Cerámica	62	5	0	67
Educación	82	9	0	91
Equipos eléctricos	162	14	0	176
Industrias manufactureras	47	1	0	48
Investigacion y desarrollo	81	9	0	90
Madera y muebles	2	1	0	3
Papel, impresión y edición	1	0	0	1
Química, plásticos	227	115	0	342
Textil	12	0	0	12
Transformados metálicos	904	112	0	1016
Vehículos a motor	2641	483	0	3124
Varios	277	75	0	352
Total	4516	830	0	5346

Tabla 9.9. Distribución de robots en España por marcas (1996) (Fuente: AER)

Marca	Hasta 1994	Δ95	Total	%
ABB	1672	468	2140	40.0
Adept	49	3	52	1.0
Comau	59	26	85	1.6
Danobat	64	5	69	1.3
DEA	38	0	38	0.7
FANUC	485	58	543	10.1
Hitachi	46	0	46	0.8
KUKA	181	53	234	4.3
Renault ACMA	197	0	197	3.7
Sepro	104	50	154	2.9
Staübli Unimation	272	12	284	5.3
Volkswagen	331	0	331	6.2
Yamaha	116	12	128	2.4
Yaskawa	220	20	240	4.5
Otros	683	102	785	15.2
Total	4516	830	5346	100

La Tabla 9.7, junto con la Figura 9.14, muestra la distribución de robots por aplicación en España a principios del año 1996. En consonancia con la tabla anterior, aquellas aplicaciones propias de la industria de automoción, como son la soldadura al arco o por puntos, son las que más robots emplean. En el próximo capítulo se explican con detalle todas las aplicaciones mencionadas en la tabla y que constituyen, con más o menos implantación, las aplicaciones más idoneas para la utilización de robots.

Por último, en la Tabla 9.9 se muestra la distribución de robots por marcas en España a comienzos del año 1996. Únicamente se mencionan aquellas con mayor índice de implantación, pues la lista global llega a incluir más de 60 marcas diferentes, muchas de ellas sin llegar a tener más de media docena de robots implantados totales.

9.6. BIBLIOGRAFÍA

[AER-96] Asociación Española de Robótica, *Estudio estadísticas de robotica* 1996, Barcelona, 1996.

[AGUAYO-90] Aguayo, F., Ortega, M. y Durán, M, "Estudio de Seguridad en Sistemas Robotizados", *Revista de Robótica y Automatización Integrada*, 47, pp 40-48, 1990.

[ANSI-92] American National Standar ANSI/RIA R15.06-1992: *Industrial Robots and Robot Systems - Safety,* 1992.

[CACI-95] CACI Products Company, *Simulation with SIMFACTORY II.5*, 1995.

[EN-91] EN 292-1, *Safety of Machinery - Basic Concepts, general principles for design*, 1991.

[ENGELBERGER-80] Engelberger, J.K., "Fundamentals and Managament", en *Robotics in Practice*. pp 3-141. Kogan Page, Londres, 1980.

[FABRIZI-87] Fabrizi, D, "La Seguridad en el Empleo del Robot, Principios y Normas Italianas e Internacionales", *Revista de Robótica,* 24, pp 41-46, 1987.

[GASCOIGNE-93] Gascoigne, A.E., "Robot Safety: A Systems Safety Approach in Overview", *Proceedings of an International Conference of the Australian Robot Association and the International Federation of Robotics,* Brisbane, Queensland, Australia, 1993.

[GROOVER-89] Groover, M.P. et al., *Robótica Industrial, Tecnología, Programación y Aplicaciones*, pp 343-593. McGraw-Hill, Madrid, 1989.

[ISO-92] International Standard ISO 10218:1992(E) *Manipulating Industrial Robot - Safety, 1992.*

[MOCHON-90] F. Mochón, *Economía. Teoría y política*, McGraw-Hill, 1990.

[RODRIGUEZ-93] Rodríguez, F.J., *Planificación automática de trayectorias libres de colisión para robots en entornos 3D*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1993.

[UNE-96] UNE-EN 775, *Robots manipuladores industriales. Seguridad*, AENOR, Marzo 1996.

[WARNECKE-82] Warnecke, H.J. y Schraft,R.D., "Application Planning", en: *Industrial Robots, Application Experience,* pp 124-257.IFS (Publications) Ltd., U.K., 1982.

Capítulo 10

APLICACIONES DE LOS ROBOTS

Si no se pisa el barro, no se hace el jarro.
Refranero popular

En la actualidad los robots se usan de manera extensa en la industria, siendo un elemento indispensable en una gran parte de los procesos de manufactura. Impulsados principalmente por el sector del automóvil, los robots han dejado de ser máquinas misteriosas propias de la ciencia ficción para ser un elemento más de muchos de los talleres y líneas de producción.

Por su propia definición el robot industrial es multifuncional, esto es, puede ser aplicado a un número, en principio ilimitado, de funciones. No obstante, la práctica ha demostrado que su adaptación es óptima en determinados procesos (soldadura, paletización, etc.) en los que hoy en día el robot es, sin duda alguna, la solución más rentable [ENGELBERGER-80], [GROOVER-89] [APPLETON-87].

Junto con estas aplicaciones ya arraigadas hay otras novedosas en las que si bien la utilización del robot no se realiza a gran escala, sí se justifica su aplicación por las condiciones intrínsecas del medio de trabajo (ambientes contaminados, salas asépticas, construcción, etc.) o la elevada exigencia en cuanto a calidad de los resultados (medicina, etc.). Estos robots se han venido llamando robots de servicio [ENGELBERGER-89].

En este capítulo se repasan las aplicaciones más frecuentes, destacando las posibilidades del robot y sus ventajas frente a otras alternativas. Asimismo, se presentan algunas de las aplicaciones más novedosas en sectores como la construcción, medicina, sector eléctrico, etc.

10.1. CLASIFICACIÓN

La *Federación Internacional de la Robótica* (IFR) estableció en 1988 una clasificación de las aplicaciones de la robótica en el sector manufacturero [TORRALVA-92]. La Tabla 10.1 presenta esta

Tabla 10.1. Clasificación de las aplicaciones industriales de la robótica

110	Manipulación en fundición
	111 Moldes
	119 Otros
130	Manipulación en moldeo de plásticos
140	Manipulación en tratamientos térmicos
150	Manipulación en la forja y estampación
160	Soldadura
	161 Al arco
	162 Por puntos
	163 Por gas
	164 Por láser
	169 Otros
170	Aplicación de materiales
	171 Pintura
	172 Adhesivos y secantes
	179 Otros
180	Mecanización
	181 Carga y descarga de máquinas
	182 Corte mecánico, rectificado, desbarbado y pulido
	189 Otros
190	Otros procesos
	191 Láser
	192 Chorro de agua
	199 Otros
200	Montaje
	201 Montaje mecánico
	202 Inserción
	203 Unión por adhesivos
	204 Unión por soldadura
	205 Manipulación para montaje
	209 Otros
210	Paletización
220	Medición, inspección, control de calidad
230	Manipulación de materiales
240	Formación, enseñanza e investigación
900	Otros

clasificación, que pretende englobar la mayor parte de los procesos robotizados en la actualidad aunque, como se ha indicado anteriormente, se pueden encontrar aplicaciones particulares que no aparecen de manera explícita en esta clasificación. En el siguiente epígrafe se tratan con detalle todas estas aplicaciones.

10.2. APLICACIONES INDUSTRIALES DE LOS ROBOTS

La implantación de un robot industrial en un determinado proceso exige un detallado estudio previo del proceso en cuestión, examinando las ventajas e inconvenientes que conlleva la introducción del robot. Será preciso siempre estar dispuesto a admitir cambios en el desarrollo del proceso primitivo (modificaciones en el diseño de piezas, sustitución de unos sistemas por otros, etc.) que faciliten y hagan viable la introducción del robot.

En cuanto al tipo de robot a utilizar, habrá que considerar aspectos de diversa índole como área de acción, velocidad de carga, capacidad de control, coste, etc. Éstos ya fueron examinados con detalle en el Capítulo 9.

Ciertas aplicaciones industriales cuentan en la actualidad con suficiente experiencia en su robotización como para poder fijar unas líneas generales en cuanto a las posibilidades reales del robot en ellas, así como en cuanto a las dificultades, cambios y ventajas a las que la introducción del robot puede dar lugar. El primer proceso robotizado, la fundición inyectada, instalado en la General Motors (Trenton) en 1960, cuenta con más de tres décadas de antigüedad, tiempo suficiente como para que se hayan valorado con detalle las ventajas e inconvenientes del robot.

A continuación van a ser analizadas algunas de las aplicaciones industriales de los robots. Dando una breve descripción del proceso, exponiendo el modo en el que el robot entra a formar parte de él, y considerando las ventajas e inconvenientes. Por último, en algunos casos, se presentará y comentará alguna aplicación práctica real.

10.2.1. Trabajos en fundición

La fundición por inyección fue el primer proceso robotizado (1960). En este proceso el material usado, en estado líquido, es inyectado a presión en el molde. Este último está formado por dos mitades que se mantienen unidas durante la inyección del metal mediante la presión ejercida por dos cilindros. La pieza solidificada se extrae del molde y se enfría para su posterior desbarbado. El molde, una vez limpio de residuos de restos de metal y adecuadamente lubricado, puede ser usado de nuevo.

En la fundición por inyección el robot puede realizar numerosas tareas auxiliares (Figura 10.1) como son:

- Extracción de las piezas del molde y transporte de éstas a un lugar de enfriado y posteriormente a otro proceso (desbarbado, corte, etc.).

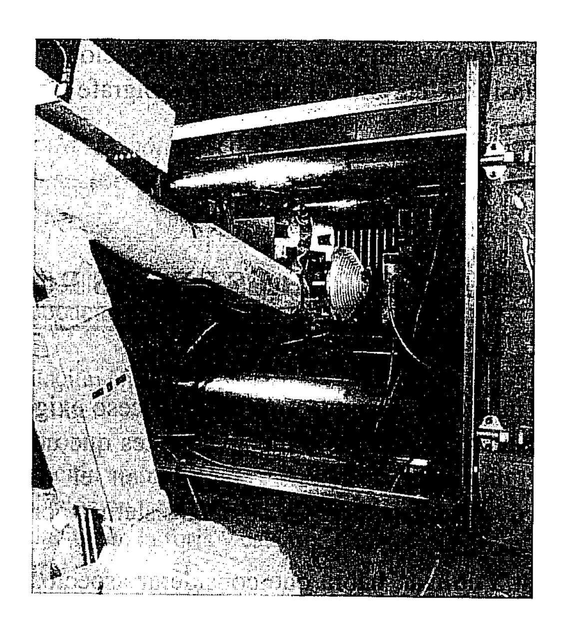


Figura 10.1. Robot MOTOMAN en una célula de fundición por inyección. (Cortesía de Yaskawa.)

- Limpieza y mantenimiento de los moldes eliminando rebabas (por aplicación de aire comprimido) y aplicando el lubricante.
- Colocación de piezas en el interior de los moldes (embutidos).

Las cargas manejadas por los robots en estas tareas suelen ser medias o altas (del orden de decenas de kilogramos), no necesitan una gran precisión (salvo tal vez si deben colocar piezas en el interior del molde) y su campo de acción ha de ser grande. Su estructura más frecuente es la polar y la articular. Su sistema de control es por lo general sencillo.

Otro proceso de fundición es el denominado fundición a la cera perdida, por microfusión o a la cáscara. Este sistema permite fundir piezas con una gran precisión y buen acabado superficial.

En este proceso, el robot puede realizar las tareas relativas a la formación del molde de material refractario a partir del modelo de cera, por sucesivas inmersiones controladas del mismo en baños de arenas refractarias. El robot, dotado de una pinza especial, recoge un racimo de modelos (conjunto de varios modelo unidos) y lo introduce en una masa de grano fino, intercalando extracciones y centrifugaciones para conseguir un reparto uniforme del recubrimiento. Varios robots en serie pueden ir introduciendo secuencialmente los racimos en diferentes tipos de arenas y, por último, en un horno, pudiéndose conseguir un proceso continuo de fabricación de moldes.

10.2.2. Soldadura

La industria automovilística ha sido la gran impulsora de la robótica industrial, empleando la mayor parte de los robots hoy en día instalados. La tarea más frecuentemente robotizada dentro de la

fabricación de automóviles ha sido sin duda alguna la soldadura de carrocerías. En este proceso, dos piezas metálicas se unen en un punto por la fusión conjunta de ambas partes, denominándose a este tipo de soldadura por puntos.

Para ello, se hace pasar una corriente eléctrica de elevada intensidad (1500 A) y baja tensión a través de dos electrodos enfrentados entre los que se sitúan las piezas a unir. Los electrodos instalados en una pinza de soldadura, deben sujetar las piezas con una presión determinada (de lo que depende la precisión de la soldadura). Además deben ser controlados los niveles de tensión e intensidad necesarios, así como el tiempo de aplicación. Todo ello exige el empleo de un sistema de control del proceso de soldadura.

La robotización de la soldadura por puntos admite dos soluciones: el robot transporta la pieza presentando ésta a los electrodos que están fijos, o bien, el robot transporta la pinza de soldadura posicionando los electrodos en el punto exacto de la pieza en el que se desea realizar la soldadura. El optar por uno u otro método depende del tamaño, peso y manejabilidad de las piezas.

En las grandes líneas de soldadura de carrocerías de automóviles, éstas pasan secuencialmente por varios robots dispuestos frecuentemente formando un pasillo. Una vez situada una carrocería en el pasillo, los robots, de manera coordinada, posicionan las pinzas de soldadura realizando varios puntos consecutivamente.

La gran demanda de robots para la tarea de soldadura por puntos ha originado que los fabricantes desarrollen robots especiales para esta aplicación que integran en su sistema de programación el control de la pinza de soldadura que portan en su extremo (Figura 10.2).

Los robots de soldadura por puntos precisan capacidad de carga del orden de los 50-100 kg y estructura articular, con suficientes grados de libertad (5 o 6) como para posicionar y orientar la pinza de

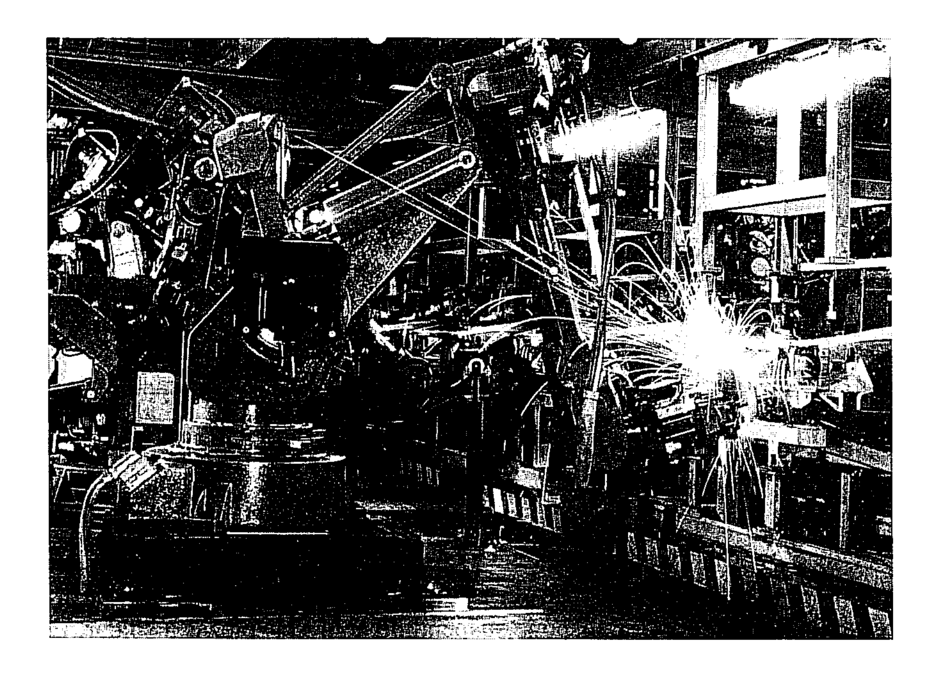


Figura 10.2. Robot IRB 6000 realizando un punto de soldadura en una carrocería de automóvil. (Cortesía ABB Robótica.)

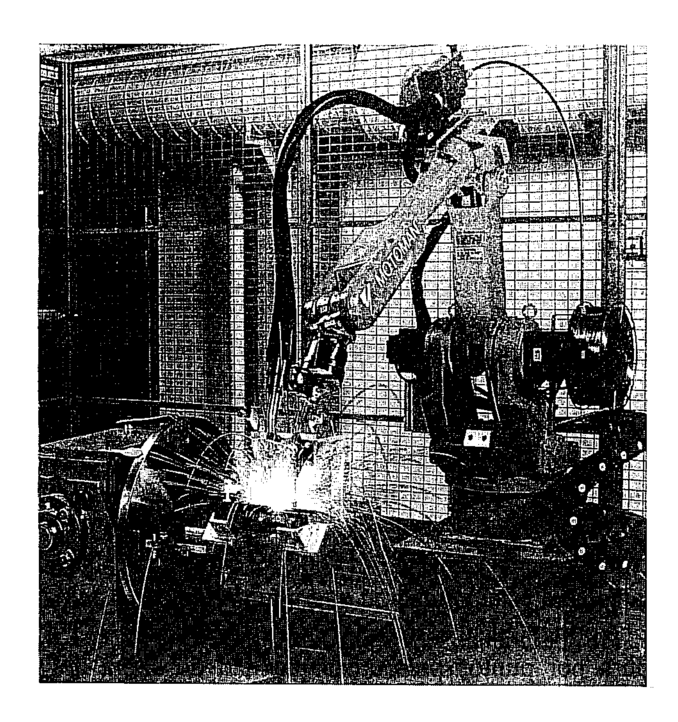


Figura 10.3. Robot MOTOMAN aplicando un cordón de soldadura. (Cortesía de Yaskawa.)

soldadura (o pieza, según el caso) en lugares de difícil acceso. En cuanto al control suele ser suficiente con un control punto a punto (PTP).

El otro tipo de soldadura, también ampliamente robotizado aunque en menor grado que el anterior, es el de soldadura por arco. En ella se unen dos piezas mediante la aportación de un flujo de material fundido procedente de una varilla de metal (electrodo). Un arco eléctrico, entre la pieza a soldar y el electrodo, origina las altas temperaturas que funden este último. La calidad de la soldadura depende de la distancia entre pieza y electrodo, de la velocidad de desplazamiento del electrodo por la trayectoria de soldadura, de la tensión e intensidad empleadas, del ángulo de ataque del electrodo, etc.

La dificultad de mantener constantes las variables antes citadas, junto con el ambiente poco agradable en el que se desarrolla la soldadura por arco (radiación ultravioleta, humo, chispas, etc.) propician que el robot sustituya al hombre en este proceso.

Los robots de soldadura por arco cuentan con una pistola que aporta el alambre del electrodo (ver Figura 10.3.). Para realizar cordones de soldadura siguiendo caminos complicados, precisan un control de trayectoria continua. No precisan gran capacidad de carga (10 kg.), pero sí un amplio campo de acción (alrededor de 2 m. de radio) y un número elevado de grados de libertad (5 o 6), incorporando incluso el control de ejes externos que mueven el utillaje que soporta a la pieza. En ocasiones, y para aumentar la accesibilidad del robot, se dispone suspendido sobre las piezas a soldar. Esto permite cubrir grandes superficies posicionando y orientando la pistola a lo largo de complicadas trayectorias.

Algunos fabricantes han dotado a las unidades de control de sus robots de instrucciones especiales para efectuar la soldadura por arco, pudiéndose especificar los parámetros de soldadura. Asimismo, la

posible imprecisión en el posicionamiento de las piezas a unir dificulta el posicionado inicial y seguimiento del cordón. Por ello, algunos robots incluyen sensores especiales para localizar el comienzo y seguir el cordón de manera automática.

Estos sistemas de seguimiento del cordón están basados normalmente en sensores láser. Este es capaz de detectar inicialmente la posición de comienzo del cordón así como de determinar su anchura para, posteriormente, seguir el cordón. Se consiguen así precisiones del orden de 0,5 mm a velocidades de 40 mm/s.

10.2.3. Aplicación de materiales. Pintura

El acabado de superficies por recubrimiento de un cierto material (pintura, esmalte, partículas de metal, etc.) con fines decorativos o de protección, es una parte crítica en muchos procesos de fabricación.

Tanto en la pintura como en el metalizado, esmaltado o arenado, la problemática a resolver es similar, siendo la primera la que cuenta con mayor difusión. Su empleo está generalizado en la fabricación de automóviles, electrodomésticos, muebles, etc.

En estos procedimientos se cubre una superficie (de forma tridimensional y en general complicada) con una mezcla de aire y material (pintura) pulverizada mediante una pistola. Es preciso conseguir una perfecta homogeneidad en el reparto de la pintura, realizándose para ello un control de la viscosidad, de la distancia entre pieza y pistola, velocidad de movimiento de ésta, número de pasadas, etc. Todos estos parámetros son tradicionalmente controlados visualmente por el operario.

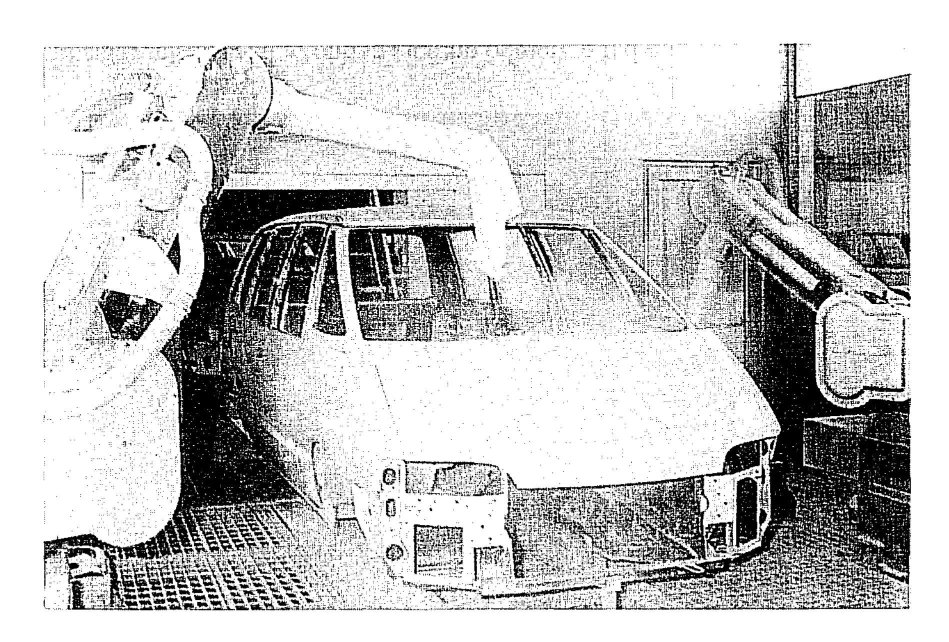


Figura 10.4. Robots en aplicación de pintura sobre la carrocería de un automóvil. (Cortesía de FANUC.)

Por otra parte el entorno en el que se realiza la pintura es sumamente desagradable y peligroso. En él se tiene simultáneamente un reducido espacio, una atmósfera tóxica, un alto nivel de ruido, un riesgo de incendio, etc. Estas circunstancias han hecho de la pintura y operaciones afines, un proceso de interesante robotización. Con el empleo del robot se eliminan los inconvenientes ambientales y se gana en cuanto a homogeneidad en la calidad del acabado, ahorro de pintura y productividad.

Normalmente los robots de pintura son específicos para este fin. Suelen ser robots articulares, ligeros, con 6 o más grados de libertad que les permiten proyectar pintura en todos los huecos de la pieza. Cuentan con protecciones especiales para defenderse de las partículas en suspensión dentro de la cabina de pintura y sus posibles consecuencias (explosiones, incendio, deterioros mecánicos, etc.). (Ver la Figura 10.4.)

Este mismo motivo origina que, en muchos casos, el accionamiento de los robots de pintura sea hidráulico, o de ser eléctrico los cables vayan por el interior de conductos a sobrepresión, evitándose así el riesgo de explosión.

Tal vez la característica fundamental de los robots dedicados a estas tareas sea su método de programación. Obviamente es preciso que cuenten con un control de trayectoria continua, pues no basta con especificar el punto inicial y final de sus movimientos, sino también el camino seguido. Sin embargo, cuando la pintura es realizada por un operario experimentado, éste no conoce el camino con precisión milimétrica, siendo cada vez diferente. Por este motivo, no es posible especificar el camino a la unidad de control del robot mediante trayectorias procedentes de una interpolación (lineal, circular, etc.). El método normal de programación de estos robots es el de aprendizaje con un muestreo continuo de la trayectoria. El operario realiza una vez el proceso de pintura con el propio robot o mediante un maniquí, mientras que la unidad de programación registra continuamente, y de manera automática, gran cantidad de puntos para su posterior repetición (esto motiva que su capacidad de memoria no se mida por el número de puntos a almacenar, sino por el tiempo durante el que se puede estar registrando puntos).

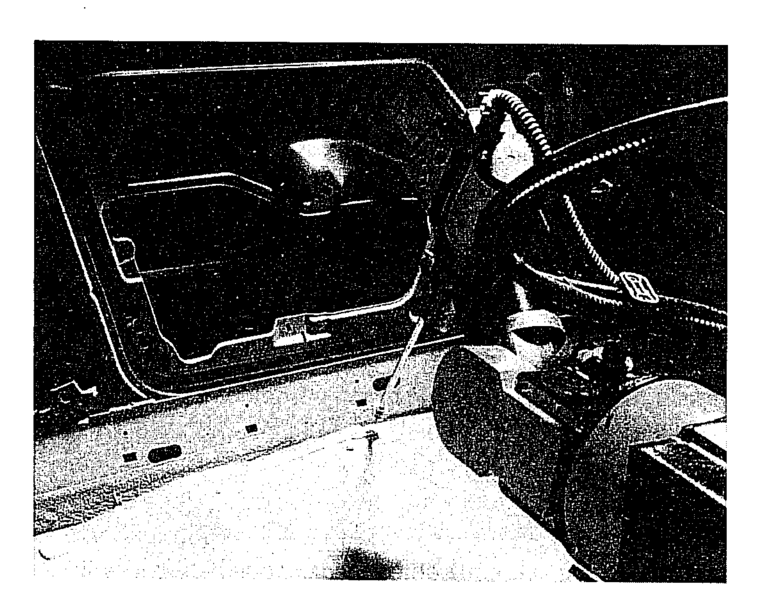


Figura 10.5. Aplicación robotizada de sellante sobre una carrocería de automóvil. (Cortesía de Kremlin.)

10.2.4. Aplicación de adhesivos y sellantes

Los robots son frecuentemente utilizados para la aplicación de cordones de material sellante o adhesivos en la industria del automóvil (sellante de ventanas y parabrisas, material anticorrosión en los bajos del coche, etc.). (Figura 10.5.)

En este proceso el material a aplicar se encuentra en forma líquida o pastosa en un tanque, siendo bombeada hasta la pistola de aplicación que porta el robot, que regula el caudal de material que es proyectado.

El robot, siguiendo la trayectoria preprogramada, proyecta la sustancia que se solidifica al contacto con el aire. En este proceso, tan importante como el control preciso de la trayectoria del robot, es el control sincronizado de su velocidad y del caudal de material suministrado por la pistola, puesto que la cantidad de material proyectado en un punto de la pieza depende de ambos factores.

Es habitual una disposición de robot suspendido sobre la pieza, siendo necesario, por los motivos antes expuestos, que el robot tenga capacidad de control de trayectoria continua (posición y velocidad regulados con precisión), así como capacidad de integrar en su propia unidad de control la regulación del caudal de material aportado en concordancia con la velocidad del movimiento.

10.2.5. Alimentación de máquinas

La alimentación de máquinas especializadas es otra tarea de manipulación de posible robotización. La peligrosidad y monotonía de las operaciones de carga y descarga de máquinas como prensas, estampadoras, hornos o la posibilidad de usar un mismo robot para transferir una pieza a través de diferentes máquinas de procesado, ha conseguido que gran número de empresas hayan introducido robots en sus talleres.

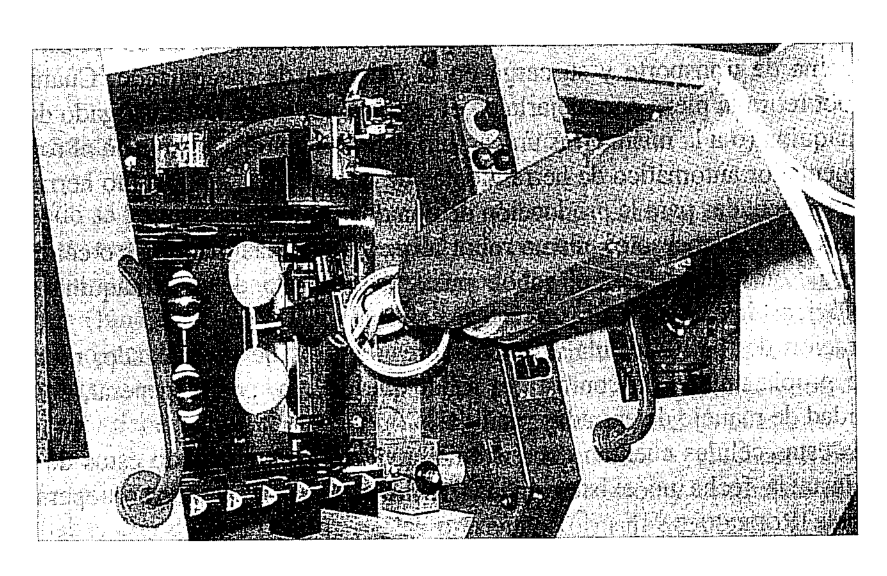


Figura 10.6. Robot IRB 2000 descargando piezas termoplásticas de una máquina de inyección. (Cortesía ABB Robótica.)

En la industria metalúrgica se usan prensas para conformar los metales en frío o, para mediante estampación y embutido, obtener piezas de complicadas formas a partir de planchas de metal. En ocasiones la misma pieza pasa consecutivamente por varias prensas hasta conseguir su forma definitiva. La carga y descarga de estas máquinas se realiza tradicionalmente a mano, con el elevado riesgo que esto conlleva para el operario, al que una pequeña distracción puede costarle un serio accidente.

Estas circunstancias, junto con la superior precisión de posicionamiento que puede conseguir el robot, y la capacidad de éste de controlar automáticamente el funcionamiento de la máquina y dispositivos auxiliares, han hecho que el robot sea una solución ventajosa para estos procesos (ver Figura 10.6).

Por otra parte, los robots usados en estas tareas son, por lo general, de baja complejidad, precisión media, número reducido de grados de libertad y un control sencillo (PTP), bastando en ocasiones con manipuladores secuenciales. Su campo de acción interesa que sea grande. En cuanto a la carga, varía mucho, pudiéndose necesitar robots con capacidad de carga de pocos kilogramos, hasta de algunos cientos (existen robots capaces de manipular hasta tonelada y media).

Las estructuras más frecuentemente usadas son la cilíndrica, esférica y articular. También la cartesiana (no pórtico) puede aportar en ocasiones la solución más adecuada.

Atención especial merece la aplicación del robot en las células flexibles de mecanizado, que han adquirido gran auge en los últimos años. Éstas emplean centros de mecanizado o varias máquinas de control numérico (CN) para conseguir complejos y distintos mecanizados sobre las piezas. En los centros de mecanizado se realiza sobre una pieza los diferentes procesos de mecanización para dar a ésta la forma programada. La capacidad de programación de estas máquinas permite una producción flexible de piezas, adaptándose así perfectamente a las necesidades del mercado actual (pequeñas y variadas series, etcétera).

Estas máquinas emplean diferentes herramientas que se acoplan a un cabezal común de manera automática cuando el proceso de mecanizado lo precisa. Las herramientas a usar en un proceso concreto son almacenadas en tambores automáticos que permiten un rápido intercambio de la herramienta.

El robot es el complemento ideal de estas máquinas. Sus tareas pueden comenzar con la recogida de la pieza del sistema de transporte y su carga en la máquina de mecanizado. Cuando esta finaliza su programa, el robot retira la pieza para dejarla en el sistema de transporte encargado de evacuarlas o para llevarla a otra máquina (o a la misma tras un cambio de orientación). Asimismo el robot puede ocuparse de cargar el alimentador automático de herramientas de la máquina, reponiendo herramientas gastadas o seleccionando las adecuadas para la producción de una determinada pieza.

En las células de multiproceso el mismo robot alimenta a varias máquinas o centros de mecanizado. Una misma pieza, transportada por el robot, puede ir pasando de una máquina a otra, incluyendo controles metrológicos de calidad u otras tareas de calibración (Figura 10.7).

La sincronización de toda la célula (alimentadores, centros de mecanizado, robots, etc.) puede ser realizada por la propia unidad de control del robot que cuenta, por lo general, con gran potencia de cálculo y capacidad de manejo de entradas y salidas.

En ocasiones estas células cuentan con sistemas multirobot, trabajando éstos de manera secuencial con la pieza. Hasta la fecha no existen apenas realizaciones prácticas de cooperación de robots de manera coordinada [RODRIGUEZ-94].

Las características de los robots para estas tareas de alimentación de máquinas herramientas son por lo general similares a las necesarias para la alimentación de otras máquinas. Las únicas discrepancias estriban en su mayor precisión y capacidad de carga inferior (decenas de kilogramo a lo sumo).

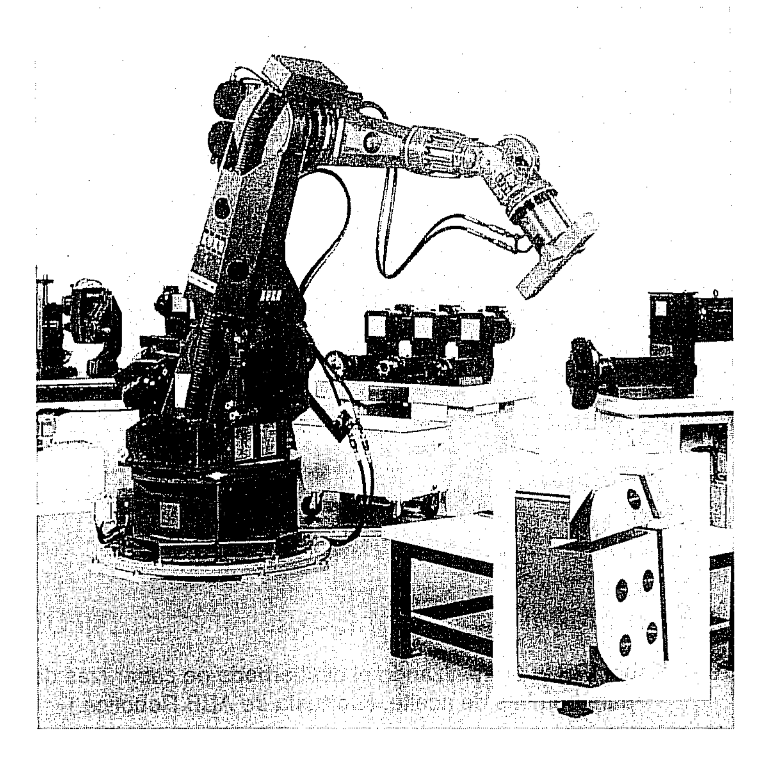


Figura 10.7. Robot IR 761/125.0 en célula de tratamiento (fresado, taladrado, serrado, etc.) de componentes de aluminio. (Cortesía de KUKA.)

10.2.6. Procesado

Dentro del procesado se incluyen aquellas operaciones en las que el robot enfrenta pieza y herramientas (transportando una u otra) para conseguir, en general, una modificación en la forma de la pieza.

El desbarbado consiste en la eliminación de rebabas de las piezas de metal o plástico, procedentes de un proceso anterior (fundición, estampación, etc.). Esta operación se realiza manualmente con una esmeriladora o fresa, dependiendo la herramienta de las características del material a desbarbar. Un robot dedicado al desbarbado porta la herramienta o la pieza, según la aplicación, haciendo entrar en contacto ambas. La herramienta debe seguir el contorno de la pieza, que en muchas ocasiones es complejo, con elevada precisión en su posicionamiento y velocidad. Por este motivo se precisan robots con capacidad de control de trayectoria continua y buenas características de precisión y control de velocidad. Además, puesto que las rebabas con que vienen las piezas presentan formas irregulares, conviene que el robot posea capacidad para adaptarse a éstas mediante el empleo de sensores o el desarrollo de un elemento terminal del robot autoadaptable (Figura 10.8).

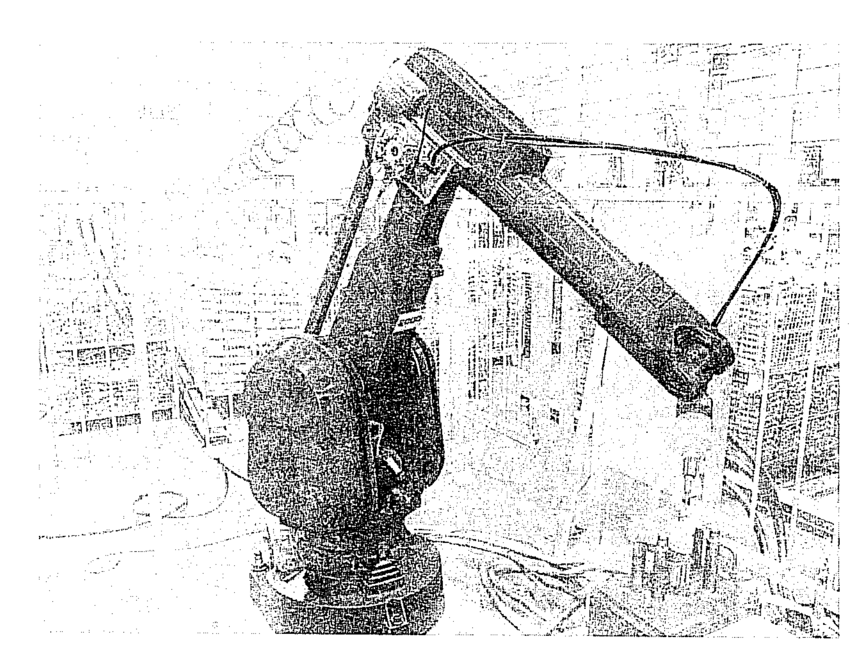


Figura 10.8. Robot IRB 2000 realizando el desbarbado de caperuzas de fundición gris para bombas de aceite. (Cortesía de ABB Robótica.)

Parecida al desbarbado en cuanto a necesidades es la aplicación de pulido, cambiando básicamente la herramienta a emplear. Las necesidades de precisión y de empleo de sensores son tal vez en este caso menos exigentes.

10.2.7. Corte

El corte de materiales mediante robot es una aplicación reciente que cuenta con notable interés. La capacidad de reprogramación del robot y su integración en un sistema CIM, hacen que aquel sea el elemento ideal para transportar la herramienta de corte sobre la pieza, realizando con precisión un programa de corte definido previamente desde un sistema de Diseño Asistido por Computador (CAD).

Los métodos de corte no mecánico más empleados son oxicorte, plasma, láser y chorro de agua, dependiendo de la naturaleza del material a cortar. En todos ellos el robot transporta la boquilla por la que se emite el material de corte, proyectando éste sobre la pieza al tiempo que sigue una trayectoria determinada (Figura 10.9).

Las piezas a cortar pueden disponerse en varias capas, unas encima de otras, realizándose el corte simultáneo de todas ellas (corte de patrones en la industria textil).

Si bien el oxicorte y el corte por plasma son tecnologías muy extendidas y consecuentemente bien conocidas, no ocurre lo mismo con el corte por láser y por chorro de agua, de más reciente aparición. La disposición típica del robot en el corte por chorro de agua es la de robot suspendido trabajando sobre las piezas fundamentalmente en dirección vertical.



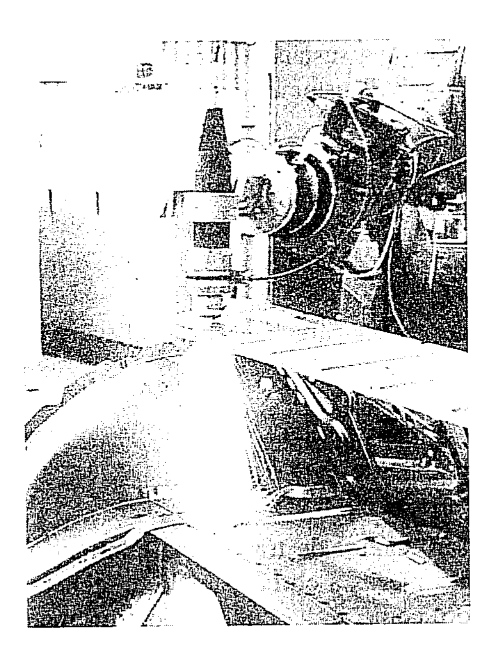


Figura 10.9. Corte por láser robotizado de planchas de aluminio. (Cortesía de KUKA.)

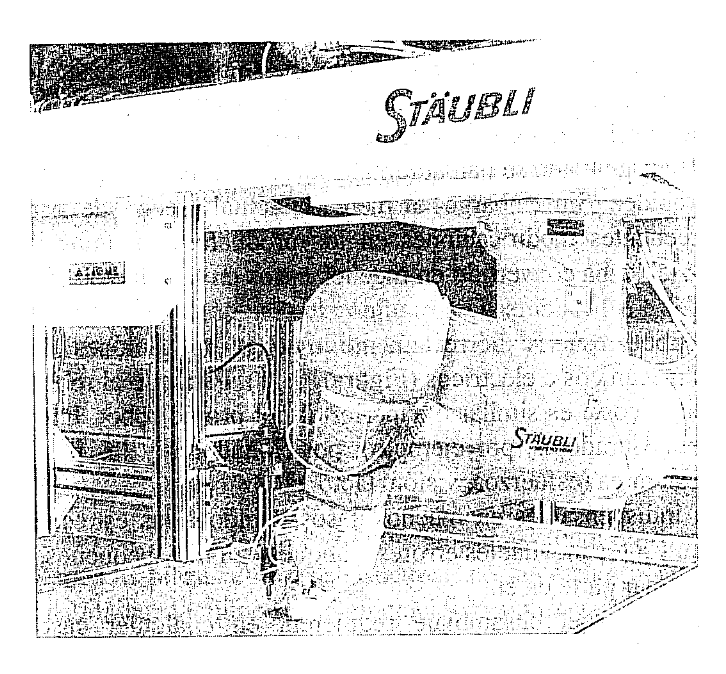


Figura 10.10. Robot Rx90 en aplicación de corte por chorro de agua. (Cortesía de Staübli.)

El robot porta una boquilla de pequeño diámetro (normalemente de 0,1 mm.) por la que sale un chorro de agua, en ocasiones con alguna sustancia abrasiva, a una velocidad del orden de 900 m/s, y a una presión del orden de 4000 kg/cm^2. El sistema completo precisa de bomba, intensificador, reguladores de presión y electroválvulas [DRAUGHTON-93].

El corte por chorro de agua puede aplicarse a materiales como alimentos, fibra de vídrio, PVC, mármol, madera, goma espuma, neopreno, yeso, tela, cartón, e incluso a metales como aluminio, acero y titanio. En estos casos se añade al agua una sustancia abrasiva.

Las principales ventajas del corte por chorro de agua frente a otros sistemas son:

- No provoca un aumento de temperatura en el material.
- No es contaminante.
- No provoca cambios de color.
- No altera las propiedades de los materiales.
- Coste de mantenimiento bajo.

Los robots empleados precisan control de trayectoria continua y elevada precisión. Su campo de acción varía con el tamaño de las piezas a cortar, siendo en general de envergadura media (de 1 a 3 metros de radio). En este sentido, como se ha comentado, con mucha frecuencia se dispone al robot suspendido boca abajo sobre la pieza (Figura 10.10).

10.2.8. Montaje

Las operaciones de montaje, por la gran precisión y habilidad que normalmente exigen, presentan grandes dificultades para su automatización flexible. Sin embargo, el hecho de que estas operaciones representen una buena parte de los costes totales del producto, ha propiciado las investigaciones y desarrollos en este área, consiguiéndose importantes avances.

Muchos procesos de ensamblado se han automatizado empleando máquinas especiales que funcionan con gran precisión y rapidez. Sin embargo, el mercado actual precisa de sistemas muy flexibles, que permitan introducir frecuentes modificaciones en los productos con unos costes mínimos. Por este motivo, el robot industrial se ha convertido en muchos casos en la solución ideal para la automatización del ensamblaje.

En particular, el robot resuelve correctamente muchas aplicaciones de ensamblado de piezas pequeñas en conjuntos mecánicos o eléctricos (Figura 10.11). Para ello el robot precisa toda una serie de elementos auxiliares cuyo coste es similar o superior al del propio robot. Entre éstos cabe destacar los alimentadores (tambores vibradores, por ejemplo), posicionadores y los posibles sensores que usa el robot para ayudarse en su tarea (esfuerzos, visión, tacto, etc.).

Estos sensores son indispensables en muchos casos debido a las estrechas tolerancias con que se trabaja en el ensamblaje y a los inevitables errores, aunque sean muy pequeños, en el posicionamiento de las piezas que entran a tomar parte en él.

Los robots empleados en el ensamblaje requieren, en cualquier caso, una gran precisión y repetitividad, no siendo preciso que manejen grandes cargas. El tipo SCARA ha alcanzado gran popularidad en este tipo de tareas por su bajo coste y buenas características. Éstas se consiguen por su *adaptabilidad selectiva*, presentando facilidad para desviarse, por una fuerza externa, en el plano

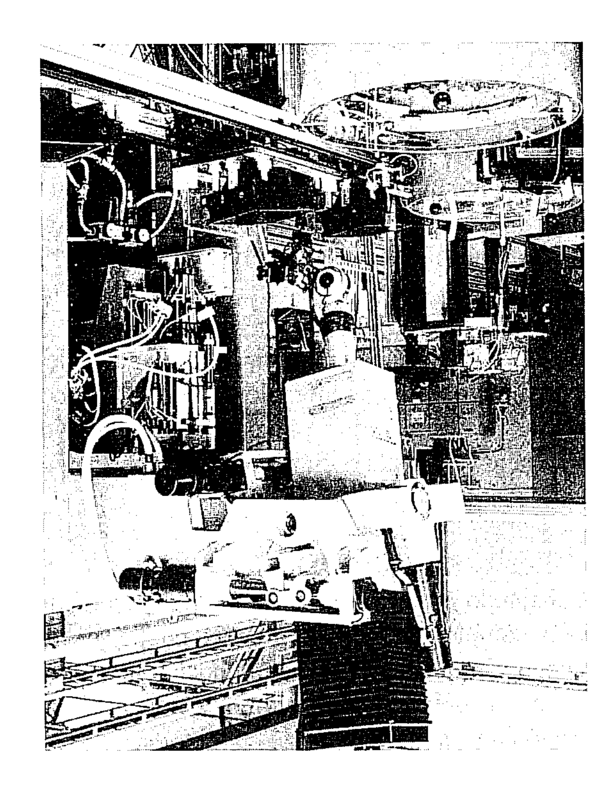


Figura 10.11. Montaje de faros de automóvil con un robot pendular tipo IRB 1000. (Cortesía de ABB Robótica.)

horizontal y una gran rigidez para hacerlo en el eje vertical. También se usan con frecuencia robots cartesianos por su elevada precisión y, en general, los robots articulares que pueden resolver muchas de estas aplicaciones con suficiente efectividad.

La dificultad inherente a este tipo de tareas obliga, en casi todos los casos, a facilitarlas con un adecuado rediseño de las partes que componen el conjunto a ensamblar. De este modo, conjuntos cuyo ensamblaje automatizado sería inabordable con su diseño inicial, pueden ser montados de una manera competitiva mediante el empleo de robots.

10.2.9. Paletización

La paletización es un proceso básicamente de manipulación, consistente en disponer piezas sobre una plataforma o bandeja (palet). Las piezas en un palet ocupan normalmente posiciones predeterminadas, procurando asegurar la estabilidad, facilitar su manipulación y optimizar su extensión. Los palets son transportados por diferentes sistemas (cintas transportadoras, carretillas, AGV, etc.) llevando su carga de piezas, bien a lo largo del proceso de fabricación, bien hasta el almacén o punto de expedición.

Dependiendo de la aplicación concreta, un palet puede transportar piezas idénticas (para almacenamiento por lotes por ejemplo), conjuntos de piezas diferentes, pero siempre los mismos

(subconjuntos procedentes de ensamblados) o cargas de piezas diferentes y de composición aleatoria (formación de pedidos en un almacén de distribución).

Existen diferentes tipos de máquinas específicas para realizar operaciones de paletizado. Estas, frente al robot, presentan ventajas en cuanto a velocidad y coste, sin embargo, son rígidas en cuanto a su funcionamiento, siendo incapaces de modificar su tarea de carga y descarga.

Así pues, los robots realizan con ventaja aplicaciones de paletización en las que la forma, número o características generales de los productos a manipular, cambian con relativa frecuencia (Figura 10.12). En estos casos, un programa de control adecuado permite resolver la operación de carga o descarga, optimizando los movimientos del robot, aprovechando al máximo la capacidad del palet o atendiendo a cualquier otro imperativo.

Generalmente, las tareas de paletización implican el manejo de grandes cargas, de peso y dimensiones elevadas. Por este motivo, los robots empleados en este tipo de aplicaciones acostumbran a ser robots de gran tamaño, con una capacidad de carga de 10 a 100 kg. No obstante, se pueden encontrar aplicaciones de paletización de pequeñas piezas, en las que un robot con una capacidad de carga de 5 kg es suficiente.

Las denominadas tareas de *pick & place*, aunque en general con características diferentes al paletizado, guardan estrecha relación con éste.

La misión de un robot trabajando en un proceso de *pick & place* consiste en recoger piezas de un lugar y depositarlas en otro. La complejidad de este proceso puede ser muy variable, desde el caso más sencillo en el que el robot recoge y deja las piezas en una posición prefijada, hasta aquellas aplicaciones en las que el robot precise de sensores externos, como visión artificial o tacto, para determinar la posición de recogida y colocación de las piezas.



Figura 10.12. Paletizado mediante robot de diversas cajas de productos. (Cortesía de FANUC.)

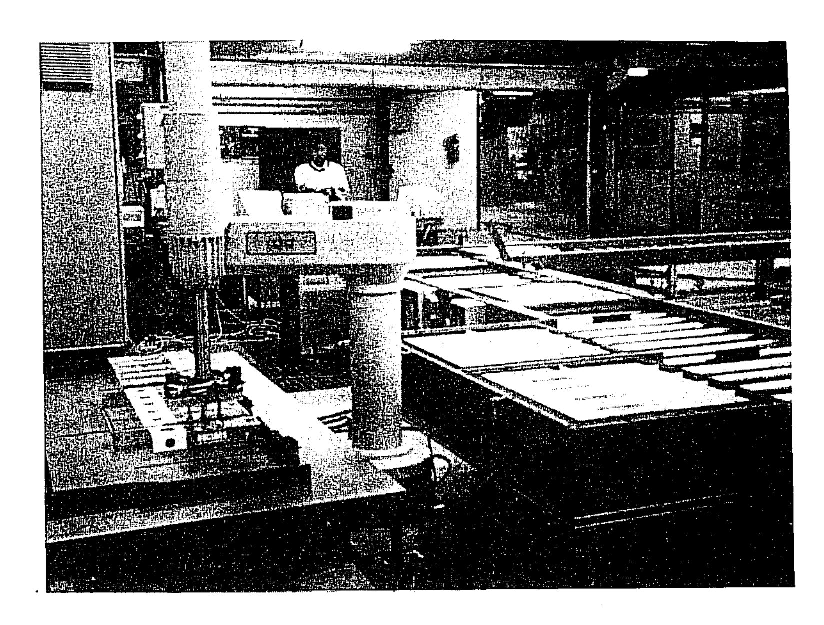


Figura 10.13. Robot AdeptThree en célula de *picking* de cartones de tabaco. (Cortesía de DISAM-UPM.)

Al contrario que en las operaciones de paletizado, las tareas de *picking* suelen realizarse con piezas pequeñas (peso inferior a los 5 kg) necesitándose velocidad y precisión.

Un ejemplo típico de aplicación del robot al paletizado sería la formación de palets de cajas de productos alimenticios procedentes de la línea de empaquetado. En estos casos, cajas de diferentes productos llegan aleatoriamente al campo de acción del robot. Allí son identificadas bien por una célula de carga, por alguna de sus dimensiones, o por un código de barras. Conocida la identidad de la caja, el robot procede a recogerla y a colocarla en uno de los diferentes palets que, de manera simultánea, se están formando. El propio robot gestiona las líneas de alimentación de cajas y de palets, a la vez que toma las decisiones necesarias para situar la caja en el palet con la posición y orientación adecuadas de una manera flexible (Figura 10.13). El robot podría ir equipado con una serie de ventosas de vacío y su capacidad de carga estaría en torno a los 50 kg.

10.2.10. Control de calidad

La tendencia a conseguir una completa automatización de la producción abarca todas las etapas de ésta, inclusive el control de calidad.

El robot industrial puede participar en esta tarea usando su capacidad de posicionamiento y manipulación. Así, transportando en su extremo un palpador, puede realizar el control dimensional de piezas ya fabricadas. Para ello el robot toca con el palpador varios puntos claves de la pieza. A partir del conocimiento que en todo instante tiene la unidad de control del robot de la posición y orientación de su extremo, se obtienen los datos relativos a la posición espacial de los puntos determinados de la pieza. Estos datos son utilizados para registrar posibles desviaciones sobre los valores deseados (Figura 10.14).

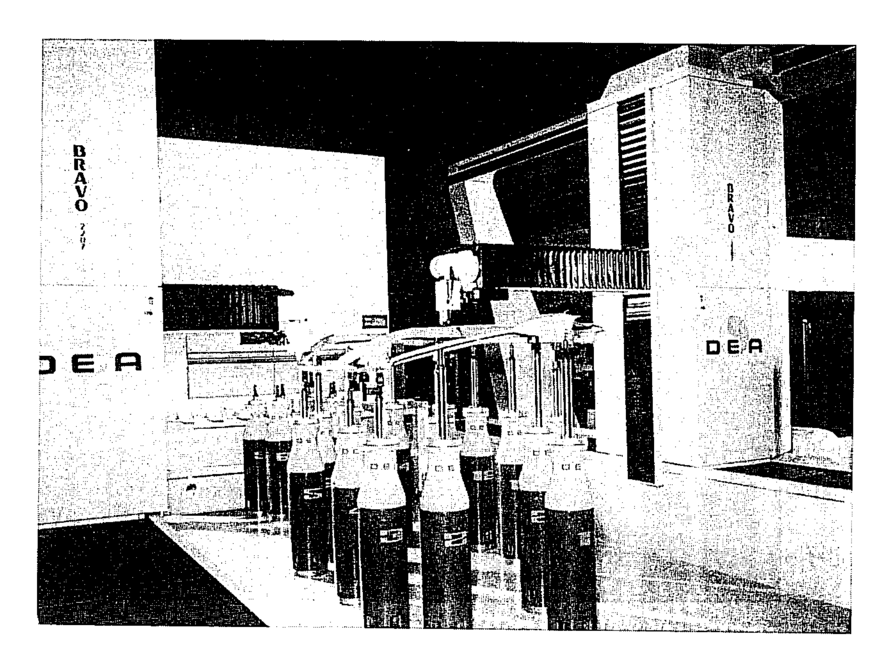


Figura 10.14. Robot cartesiano BRAVO 2207 realizando el control dimensional del lateral de un automóvil. (Cortesía de DEA.)

Otras posibles aplicaciones del robot en el control de calidad consisten en utilizar a éste para transportar el instrumental de medida (ultrasonidos, rayos X, etc.) a puntos concretos de la pieza a examinar. La situación de posibles defectos detectados puede registrarse y almacenarse a partir, como antes, de la propia unidad de control de robot (Figura 10.15).

Por último, el robot puede usarse como mero manipulador encargado de clasificar piezas según ciertos criterios de calidad (piezas correctas e incorrectas, por ejemplo). En este caso, el control y decisión de a qué familia pertenece la pieza se hace mediante un sistema específico, capaz de comunicarse con el robot (visión artificial, sistema de pesaje, etc.).

No existe, en este caso, un tipo concreto de robot más adecuado para estas tareas. En el control dimensional suelen usarse robots cartesianos por la precisión de éstos pero, en general, son igualmente válidos robots articulares.

10.2.11. Manipulación en salas blancas

Ciertos procesos de manipulación deben ser realizados en ambientes extremadamente limpios o/y controlados. En ellos, la actividad del operador se ve dificultado no por el trabajo en sí, que no tiene porqué ser especialmente complejo o delicado, sino por la necesidad de mantener elevadas medidas de control de impurezas mediante el uso de trajes especiales y controles rigurosos. Las denominadas salas blancas de la industria de los semiconductores o las salas de fabricación de algunos productos farmacéuticos, son ejemplos típicos.

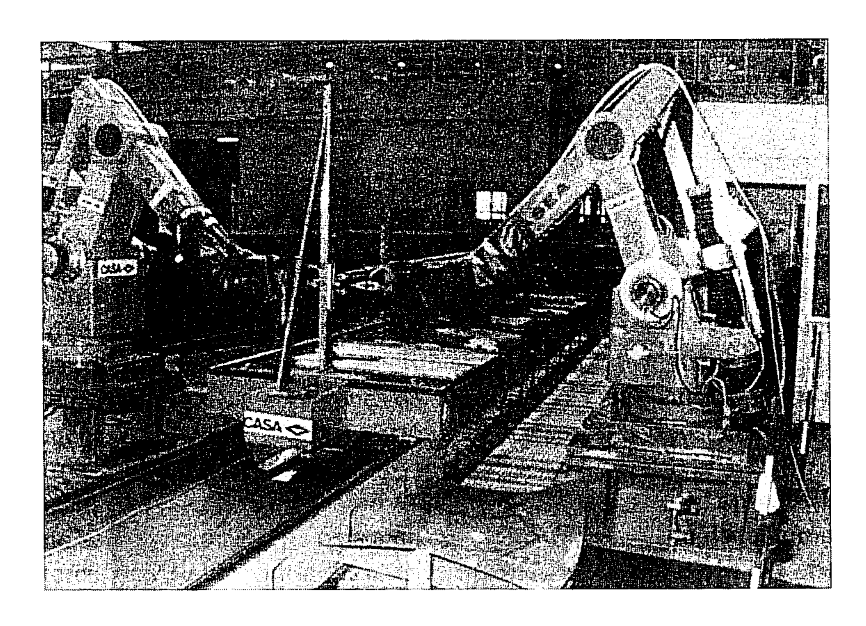


Figura 10.15. Robots IRB 60 realizando un control de calidad por ultrasonidos. (Cortesía de DISAM-UPM.)

Diversos procesos de la industria farmacéutica, como la producción de vacunas y hormonas, o la preparación de injertos de piel y reproducción de células, deben ser realizadas bajo estrictas condiciones de esterilidad. La manipulación de estos productos durante su fabricación se realiza en cabinas con protección de clase 10, en las que los operadores deben pasar por un minucioso proceso de esterilización antes de entrar al interior.

Las funciones típicas de manipulación a realizar en estos procesos son la recogida de una de las probetas que contienen el producto en fase de fabricación, su apertura, la adición de algún nuevo producto, giro de la probeta, etc.

La utilización de un robot para estas funciones se realiza introduciendo éste de manera permanente en la cabina. El robot debe cumplir la normativa correspondiente al entorno de clase 10, siendo por lo demás válido cualquier robot comercial, normalmente de seis grados de libertad y alcance inferior a un metro. De este modo se consigue, entre otros beneficios, una reducción del riesgo de contaminación, una mayor homogeneidad en la calidad del producto, y una reducción en el coste de la fabricación (ver Figura 10.16).

Con menos exigencias de limpieza que las demandadas en la fabricación de semiconductores o en la industria farmacéutica, la industria alimenticia puede hacer uso de robots para la preparación o procesado de alimentos. La preparación de bandejas de comida (*catering*), la preparación de cajas de hamburguesas (Figura 10.17) o incluso el corte de piezas de carne en un matadero, son algunos ejemplos de sistemas hoy en día en funcionamiento.

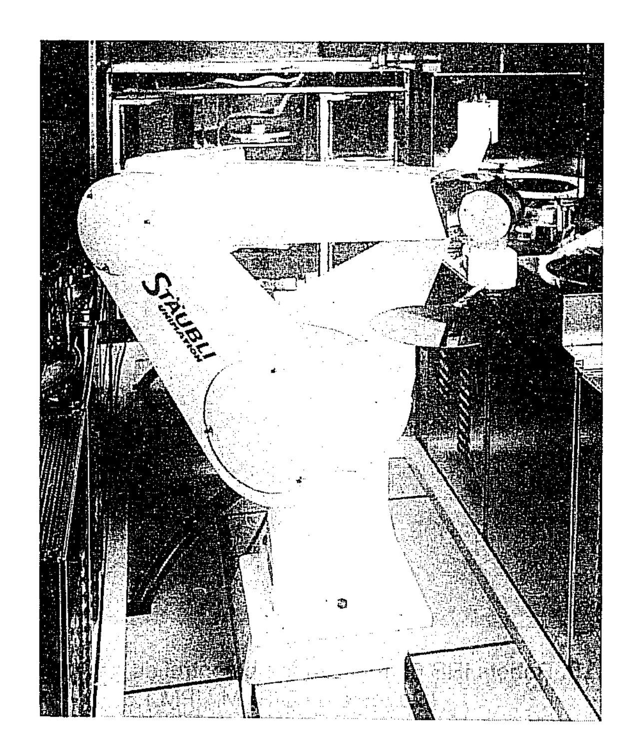


Figura 10.16. Manipulación de obleas en una sala blanca. (Cortesía de Staübli.)

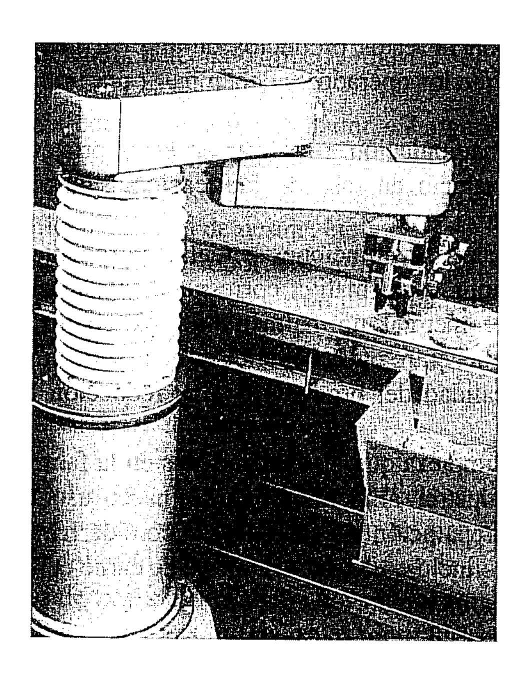


Figura 10.17. Preparación robotizada de cajas de hamburguesas en la industria alimentaria. (Cortesía de FANUC.)

10.3. NUEVOS SECTORES DE APLICACIÓN DE LOS ROBOTS. ROBOTS DE SERVICIO

Las aplicaciones de la robótica examinadas anteriormente responden a los sectores que, como el del automóvil o de la manufactura, han sido desde hace 30 años usuarios habituales de los robots industriales. Este uso extensivo de los robots en los citados sectores se ha visto propiciado por la buena adaptación del robot industrial a las tareas repetitivas en entornos estructurados. De este modo, la competitividad del robot frente a otras soluciones de automatización se justifica por su rápida adaptación a series cortas, sus buenas características de precisión y rapidez, y por su posible reutilización con costes inferiores a los de otros sistemas.

Sin embargo, existen otros sectores donde no es preciso conseguir elevada productividad, en los que las tareas a realizar no son repetitivas, y no existe un conocimiento detallado del entorno.

Entre estos sectores podría citarse la industria nuclear, la construcción, la medicina o el uso doméstico. En ninguno de ellos existe la posibilidad de sistematizar y clasificar las posibles aplicaciones, pues éstas responden a soluciones aisladas a problemas concretos.

Este tipo de robots ha venido a llamarse **robots de servicio** y pueden ser definidos como [NOSTRAND-90]:

- Un dispositivo electromecánico, móvil o estacionario, con uno o más brazos mecánicos, capaces de acciones independientes.

Estos robots están siendo aplicados en sectores como:

- Agricultura y silvicultura.
- Ayuda a discapacitados.
- Construcción.
- Domésticos.
- Entornos peligrosos.
- Espacio.
- Medicina y salud.
- Minería.
- Submarino.
- Vigilancia y seguridad.

En general, la aplicación de la robótica a estos sectores se caracteriza por la falta de estructuración tanto del entorno como de la tarea a realizar, y la menor importancia de criterios de rentabilidad económica frente a la de realizar tareas en entornos peligrosos o en los que no es posible el acceso de las personas.

Estas características obligan a que los robots de servicio cuenten con un mayor grado de *inteligencia*, que se traduce en el empleo de sensores y del software adecuado para la toma rápida de decisiones. Puesto que en muchas ocasiones el estado actual de la inteligencia artificial (disciplina que aborda esta problemática) no está lo suficientemente desarrollado como para resolver las situaciones planteadas a los robots de servicio, es frecuente que éstos cuenten con un mando remoto, siendo en muchas ocasiones robots teleoperados.

Centros de investigación en robótica, como la Universidad Carnegie-Mellon o el Jet Propulsion Laboratory (JPL) en Estados Unidos, han orientado desde hace tiempo buena parte de sus esfuerzos de investigación en robótica en esta línea, desarrollando robots especializados, capacitados para trabajar en el exterior, en entornos no estructurados y peligrosos (superficie de planetas, volcanes, desastres nucleares, etc.).

Muchos de estos sistemas responden en buena medida a los conceptos de telemanipulador o robot teleoperado, que quedan fuera de los objetivos de este texto, remitiéndose al lector a textos especializados [SHERIDAN-92].

Para dar una idea general de las posibilidades de la robótica en estos sectores, se van a comentar algunas aplicaciones concretas en las que el robot ha aportado mejoras y beneficios al proceso primitivo.

10.3.1. Industria nuclear

Por sus especiales características, el sector nuclear es uno de los más susceptibles de utilizar robots de diseño específico [FOGLE-93]. Entre las diversas aplicaciones se han escogido aquí, por su especial relevancia, las relativas a operaciones de mantenimiento en zonas contaminadas y de manipulación de residuos.

Inspección de los tubos del generador de vapor de un reactor nuclear

Las operaciones de inspección y mantenimiento de las zonas más contaminadas de una central nuclear de producción de energía eléctrica son por naturaleza largas y costosas.

De realizarlas manualmente, el tiempo de exposición de los operadores a la radiación es un factor crítico que, junto con el elevado coste que supone una interrupción temporal del funcionamiento del sistema en cuestión, justifica sin ningún lugar a dudas la utilización de sistemas robotizados, normalmente teleoperados, total o parcialmente, que sustituyan al operador.

En el generador de vapor se produce el intercambio de calor entre el fluido primario y secundario. Para ello, dentro de la vasija del generador, se encuentran dispuestas en forma matricial los tubos por los que circula el fluido receptor del calor.

El inevitable desgaste de estos tubos obliga a realizar periódicamente labores de inspección, para que en el caso de que alguno se encuentre dañado inutilizarlo, poniendo en funcionamiento alguno de los tubos de reserva que a tal fin se han dispuesto en el generador.

Para realizar esta operación de manera automática puede utilizarse un robot de desarrollo específico que, introducido en la vasija, posicione una sonda de inspección en la boca de cada tubo. Ésta, empujada por el interior del tubo, proporcionará información sobre el estado del mismo.

Es preciso considerar que el robot se introduce en la vasija mediante un sistema mecánico que, junto con los posibles errores en la disposición matricial de los tubos, obliga a que el robot trabaje, bien con

ayuda de teleoperación, o bien con sistemas sensoriales externos como visión o láser, que proporcionen la posición real relativa entre el extremo del robot y los tubos.

Manipulación de residuos radioactivos

Como se indicó en el Capítulo 1, los primeros sistemas teleoperados, desarrollados por el Argonne National Laboratory (ANL) tuvieron como objetivo la manipulación de productos radioactivos sin riesgo para el ser humano.

La industria nuclear genera una cantidad considerable de residuos radioactivos de baja contaminación (vestimentas, envases de plástico, papel, etc.) o de alta (restos de las células del reactor, materiales en contacto directo prolongado con las zonas radiactivas, etc.). La forma, tamaño y peso de estos desechos es variable y su manipulación tiene por objetivo final su envase en contenedores especiales, que son posteriormente transportados y almacenados (lo que origina una nueva problemática).

Para manipular remotamente estos residuos se hace uso tanto de telemanipuladores con unión mecánica y seguimiento directo del proceso por parte del operador a través de un cristal (caso de baja contaminación), como con sistemas con mando remoto por radio o cable en el caso de contaminación elevada. Estos manipuladores permiten la flexibilidad necesaria para manipular elementos de peso variable y forma no definida (incluso no rígidos).

Además, es preciso considerar la importancia que tiene la optimización del espacio ocupado por los residuos en su almacenamiento, por lo que antes de su envasado en los contenedores puede ser preciso fragmentarlos [LARCOMBE-84].

10.3.2. Medicina

De entre las varias aplicaciones de la robótica a la medicina destaca la cirugía. Las primeras aplicaciones de la robótica a la cirugía del cerebro datan del año 1982. En esta fecha se comenzó en Memorial Medical Center de Long Beach (California) un programa cuyo objetivo consistía en utilizar un robot comercial (PUMA 260) para realizar determinadas operaciones de neurocirugía. Desde entonces se han puesto a punto varios sistemas que, con la ayuda de un *scanner*, un ordenador registre toda la información necesaria del cerebro para que el equipo médico decida el punto exacto donde debe ser realizada la incisión, donde penetrará la sonda para obtener una muestra para realizar una biopsia. El robot, que se encuentra perfectamente situado con respecto al paciente, porta en su extremo los instrumentos necesarios para realizar la incisión, tomar la muestra, etc.

La utilización de un robot conectado al ordenador permite que tanto la incisión como la toma de la muestra se realicen con la máxima precisión y en un tiempo notablemente inferior al que se consumiría en caso de emplear el sistema habitual.

Además, se descarga al cirujano de la mecánica de ciertas tareas como el correcto posicionamiento de los instrumentos de cirugía con respecto al cráneo del paciente, permitiendo una mayor concentración en el seguimiento y control de la operación.

También, otro posible beneficio de la aplicación de la robótica a la cirugía se encuentra en el telediagnóstico y la telecirugía. Esta última consiste en la operación remota de un paciente mediante un telemanipulador.

En julio de 1993 un robot SCARA en Milán (Italia) realizó sobre el hígado de un cerdo una biopsia y una incisión para introducir un laparoscopio. El robot estaba siendo teleoperado en tiempo real desde

14000 km de distancia en el Jet Propulsion Lab de Pasadena, California. Las órdenes del cirujano y la información procedente de la sala de operaciones eran intercambiadas a través de 2 satélites de comunicaciones y redes de fibra óptica.

Esta primera experiencia en operación remota permite asegurar que este tipo de intervenciones sobre pacientes humanos podrá ser una realidad en un futuro próximo. Su justificación puede encontrarse en el tratamiento de pacientes situados en localizaciones difícilmente alcanzables (espacio, plataformas submarinas, escenarios bélicos, etc.). No obstante hay que considerar que aún existen importantes dificultades técnicas, como es el retraso en las comunicaciones (en el caso mencionado del orden de varios segundos) y el elevado coste (es preciso, por ejemplo, el uso exclusivo de los satélites de comunicaciones).

10.3.3. Construcción

El sector de la construcción es, en la mayoría de los países industrializados, uno de los que moviliza mayor número de recursos económicos y humanos. No es pues de extrañar que desde hace algo más de una década se estén desarrollando gran número de sistemas robotizados, orientados a automatizar en lo posible algunas de las múltiples labores que entran a formar parte de los procesos constructivos.

En este tipo de aplicaciones de la robótica, como en otros muchos, es Japón el país que cuenta con un mayor número de sistemas en funcionamiento. En algunos casos se trata de robots parcialmente teleoperados, construídos a partir de maquinaria convencional (gruas, excavadoras, etc.). En otros es maquinaria específicamente construída para resolver un proceso concreto.

Si se analizan las condiciones existentes para la robotización de la construcción se llega entre otras a las siguientes conclusiones [PUENTE-95]:

- Las condiciones de trabajo son complejas.
- Los robots deben tener capacidad de locomoción y cierto grado de inteligencia.
- Los robots deben estar preparados para trabajar en exteriores, moviéndose en entornos difíciles y no protegidos.
- Deben manejar piezas pesadas y de grandes dimensiones.
- Las operaciones a realizar son complejas, variadas y poco repetitivas.
- Los robots deben ser fácilmente transportables a la obra.

Con estos condicionantes, las posibles tareas robotizables dentro de la construcción de edificios (comerciales, industriales o residenciales) podrían agruparse en [OPPENHEIM-90]:

1. Operaciones de colocación de elementos.
   - Construcción mediante colocación repetitiva de estructuras básicas (ladrillos, bloques, etc.).
   - Posicionamiento de piezas, normalmente grandes y pesadas (vigas, etc.).
   - Unión de diferentes piezas que componen una estructura (soldadura, remaches, etc.).
   - Sellado de las uniones entre diferentes piezas.

2. Operaciones de tratamiento de superficies.
   - Acabado de superficies (pulido, etc.).
   - Recubrimiento de superficies con pintura, barniz, etc.
   - Extensión de material sobre la superficie (cemento, espuma aislante, etc.).

3. Operaciones de rellenado.
   - Vertido de cemento u hormigón en encofrados.
   - Excavación para la preparación de terrenos y movimiento de tierras.
   - Rellenado con tierra de volúmenes vacios.

4. Otras.
   - Inspección y control.

Tabla 10.2. Aplicación de la robótica en la industria de la construcción

Denominación	Actividad	Empresa	País	Estado
Surf-Robo	Suelos de cemento	Takenaka Corp.	Japón	En uso
...	Soldadura de columnas	Shimizu Corp.	Japón	En uso
...	Túneles	Hitachi Zosen Corp.	Japón	En uso
LAMA	Limpieza de fachadas	Putzmeister	Alemania	Prototipo
Blockbot	Cerramiento de ladrillos	MIT	USA	Laboratorio
GRC-Auto	Prefabricados de GRC	DISAM-Dragados	España	En uso
CM-14	Extracción de material	Carnegie-Mellon Univ.	USA	Prototipo
WA150	Asfaltado de carreteras	Nippon Hodo Co. Komatsu Ltd.	Japón	Prototipo
ROCCO	Cerramientos y tabiques interiores de edificios	DISAM-Karlsruhe Univ.	España-Alemania	Prototipo

Fuera de la construcción de edificios, cabrían destacar las realizaciones de robots para la construcción de túneles, carreteras (asfaltado), inspección de estructuras en puentes o muros de edificios difícilmente accesibles, etc.

La Tabla 10.2 recoge alguna de las realizaciones más conocidas en este área [GAMBAO-96]. Como se puede observar, destacan los desarrollos japoneses. No obstante, debe indicarse que, como en otras muchas ocasiones, los robots japoneses son, en su mayoría, teleoperados total o parcialmente, mientras que en Europa y EE. UU. existe una mayor tendencia al desarrollo de sistemas con un mayor grado de autonomía (Figura 10.18).

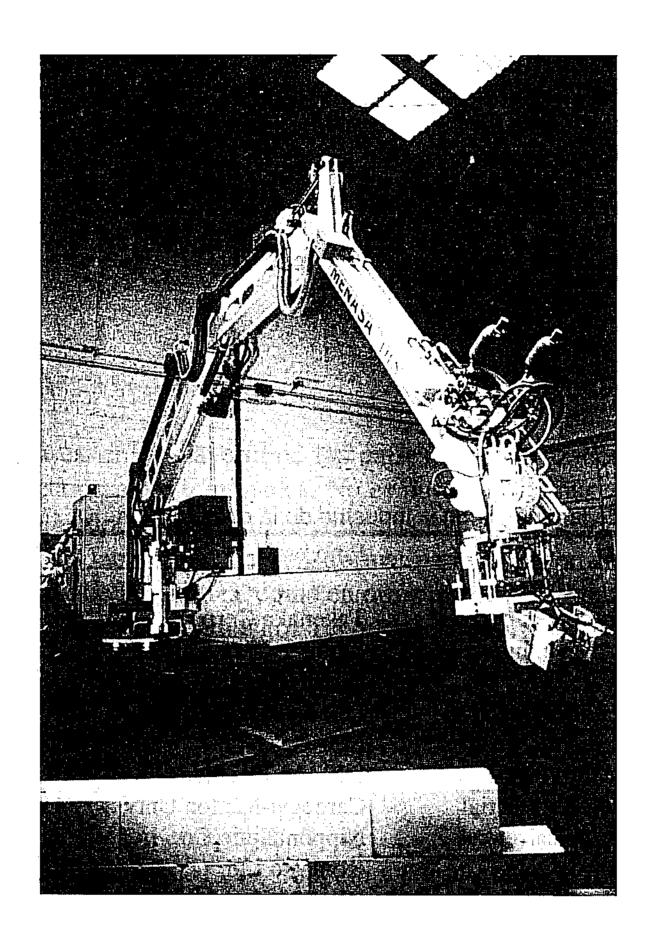


Figura 10.18. Robot ROCCO para la construcción mediante el posicionamiento automático de bloques de hormigón. (Cortesía DISAM-UPM.)

10.4. BIBLIOGRAFÍA

[APPLETON-87] E. Appleton y D.J. Williams, *Industrial Robot Applications*, Halsted Press, John Wiley and Sons y Open University Press, 1987.

[DRAUGHTON-93] R.N. Draughton, "Robotic ultrahigh-pressure waterjet cutting", en: *Robotics for Competitive Industries*, Mechanical Engineering Publications Limited, London, 1993.

[ENGELBERGER-89] J.F. Engelberger, *Robotics in Service*, Kogan Page, Londres, 1989.

[ENGELBERGER-80] J.F. Engelberger, *Robotics in Practice*, Kogan Page, Londres, 1980.

[FOGLE-93] Fogle, R.F., "Robots and teleoperators in radioactive applications", en: *Robotics and Remote Systems for Hazardous Environments*, Ed. M.Jamshidi y P.J. Eicker, Prentice-Hall, 1993.

[GAMBAO-96] E. Gambao, *Control de robots con accionamientos hidráulicos. Aplicación a robots de grandes dimensiones para la construcción*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid.

[GROOVER-89] M.P. Groover et. al , "Aplicaciones de robot en fabricación", en: *Robótica industrial. Tecnología, programación y aplicaciones*, McGraw-Hill, Madrid, 1989.

[LARCOMBE-84] Larcombe y J.R. Halsall, *Robotics in Nuclear Engineering*, C.E.C., EUR 9312EN, 1994.

[NOSTRAND-90] J.W. Nostrand y E.L. Sampson, "Robots in Service Industry", en: *Concise International Encyclopledia of Robotics*, Editor R.C. Dorf, John Wiley & Sons , 1990.

[OPPENHEIM-90] J.J. Oppenheim, "Robots in Construction", en: *Concise International Encyclopedia of Robotics*,(Editor. R.C. Dorf), John Wiley and Sons, 1990.

[PUENTE-95] E.A. Puente, *Nuevos robots para nuevas aplicaciones. Robótica en la construcción civil*, documento interno de DISAM, Universidad Politécnica de Madrid, 1995.

[RODRIGUEZ-94] Rodríguez, C.F., *Control y planificación de sistemas multirrobot trabajando en tareas coordinadas*, Tesis Doctoral, DISAM, Universidad Politécnica de Madrid, 1994.

[SHERIDAN-92] Sheridan, T.B., *Telerobotics, Automation and Human Supervisory Control*, The MIT Press, 1992.

[TORRALVA-92] Torralva, C., "Nueva clasificación de la IFR para los robots", *Automatización industrial y revista de robótica*, 67, Abril 1992.

Índice analítico

"Ale do desíti let nadêlají Rossumovi Universální Roboti tolik
psenice, tolik látek, tolik vseho, ze rekneme:
vêci uz nemají ceny. Nyní ber kazdý, kolik potrebujes.
Není bídy. Ano, budou bez práce. Ale pak nebude uz vubec zádné práce.
Vsechno udêlají zivé stroje. Clovêk bude dêlat jen to, co miluje.
Bude zít jen proto, aby se zdokonaloval."

("Pero dentro de diez años Robots Universales Rossum
producirá tanto trigo, tantos tejidos, tanto de todo,
que las cosas prácticamente carecerán de valor.
Cada cual podrá coger lo que quiera. No habrá pobreza.
Sí habrá desempleo, pero no habrá empleo.
Todo lo harán máquinas vivientes. Todo el mundo estará
libre de preocupaciones y del degradante trabajo manual.
Todos vivirán solo para perfeccionarse.")

Domin, primer acto de "R.U.R"
Karel Capek